Review

Using Opinion Mining in Context-Aware Recommender Systems: A Systematic Review

Camila Vaccari Sundermann [1,*], **Marcos Aurélio Domingues** [2]**, Roberta Akemi Sinoara** [1]**, Ricardo Marcondes Marcacini** [3] **and Solange Oliveira Rezende** [1]

[1] Institute of Mathematics and Computer Science—University of São Paulo, Avenida Trabalhador São Carlense, 400, São Carlos SP 13566-590, Brazil; rsinoara@usp.br (R.A.S.); solange@icmc.usp.br (S.O.R.)

[2] Department of Informatics—State University of Maringá, Avenida Colombo, 5790, Maringá PR 87020-900, Brazil; madomingues@uem.br

[3] Federal University of Mato Grosso do Sul—Três Lagoas Campus, Avenida Ranulpho Marques Leal, 3484, P.O. Box 210, Três Lagoas MS 79620-080, Brazil; ricardo.marcacini@ufms.br

[*] Correspondence: camilavs@icmc.usp.br

Abstract: Recommender systems help users by recommending items, such as products and services, that can be of interest to these users. Context-aware recommender systems have been widely investigated in both academia and industry because they can make recommendations based on a user's current context (e.g., location and time). Moreover, the advent of Web 2.0 and the growing popularity of social and e-commerce media sites have encouraged users to naturally write texts describing their assessment of items. There are increasing efforts to incorporate the rich information embedded in user's reviews/texts into the recommender systems. Given the importance of this type of texts and their usage along with opinion mining and contextual information extraction techniques for recommender systems, we present a systematic review on the recommender systems that explore both contextual information and opinion mining. This systematic review followed a well-defined protocol. Its results were based on 17 papers, selected among 195 papers identified in four digital libraries. The results of this review give a general summary of the current research on this subject and point out some areas that may be improved in future primary works.

Keywords: recommender systems; context-aware recommender systems; contextual information; opinion mining; systematic review

1. Introduction

Nowadays, with the growth of the digital universe, e-commerce, and social networks, a great diversity of information, products, and services is available on the Web. Users find, while browsing, many news, products, movies and people in the social networks. With so many options, the big challenge is to identify what is really relevant that meets the real interests and preferences of users. Thus, recommender systems have emerged with the purpose of assisting users in their choices. A recommender system is an information filtering technology that can be used to predict ratings for items (products, services, movies, among others), and/or generate a custom item ranking which may be of interest to the target user [1]. In this way, this type of system can aid in decisions like "which product to buy", "which movie to watch" and "which hotel to book".

One of the main domains that currently use recommender systems is the e-commerce, in which sites interact directly with customers suggesting products of interest with the aim of increasing their sales. For example, the Amazon website [2], which was one of the precursors in this area, makes recommendations to users in the form: "Customers who bought this item also bought ..." or

"Customers who viewed this item also viewed ..." [3]. Sites from various domains such as Netflix [4], Last.fm [5], TripAdvisor [6] and Facebook [7] also use recommender systems. The use of such systems can represent a considerable competitive advantage on the Web.

Traditional recommender systems focus on user and item data to generate recommendations. Examples of such techniques include collaborative filtering, content-based filtering, and hybrid approaches. Collaborative filtering is a recommendation technique that finds correlations among users or among items to generate recommendations [8]. This technique aims to make recommendations based on items already evaluated by other users with behaviors similar to the target user, or based on items with similar rating pattern to the items already evaluated by the current user. Content-based filtering techniques make recommendations considering the attributes of items belonging to the user's access history [9]. In addition, the hybrid approaches combine both techniques. These techniques, in their traditional forms, consider only the set of ratings or user accesses. However, empirical studies indicate that context-aware approaches can produce more precise recommendations [10–17]. A travel package recommender system, for example, can improve the performance of the recommendation by considering the context "season of the year" in which the user wishes to travel since some places are recommended in "summer" while others are recommended in "winter". There are many definitions of context in the literature, depending on the application area [1]. In this work, the term context is defined as any information that can be used to characterize the situation of an entity (item or user) [18].

Context-aware recommender systems have been widely investigated both in the academic field and in companies [19]. However, some challenges are still faced by this type of system. One of the main challenges is the difficulty in the acquisition of contextual information. There is a lack of automatic methods for extracting this type of information. In this way, effective methods and strategies for identifying contextual information are investigated. On the other hand, with the advancement of Web 2.0 and the growing popularity of social networking and e-commerce, users have been increasingly encouraged to write about their opinions on the items, such as reviews. Reviews are usually in the form of textual comments, in which users, based on their experiences, explain why they liked or disliked certain items. There is an effort to incorporate the important information that can be extracted from reviews into recommender systems. According to Chen and Chen [20], this information can aid recommender systems in the following ways: (i) it can help to solve the problem of data sparsity; and (ii) it can help to solve the cold-start problem.

In addition, according to Chen and Chen [20], some of the rich information that can be extracted from reviews and which can be used to improve the recommendation performance are: (i) aspects, which are the attributes of an item that a user discusses in a review; (ii) overall opinions, which can be represented by the orientations of users' sentiments for items; (iii) aspect opinions, which are opinions about specific characteristics of an item; and (iv) contextual information, since, according to Hariri et al. [12], users usually provide some context hints in their comments.

Thus, reviews or another type of user-generated text can provide relevant information for recommender systems, such as contextual information and opinions. For example, a user can mention in reviews that he/she stayed at a particular hotel during a trip with his/her family or during a business trip and can express his/her opinions about hotel services that were important to him/her in that particular context. In the following review, extracted from the site *TripAdvisor*, presented in Chen and Chen [20] and adapted as an example for this work, it is possible to observe the opinions of the hotel aspects (in bold) and the contextual information (underlined):

> "First trip to Asia, first visit to company's Hong Kong offices and the Four Seasons HK provided a great base for all of it. **Rooms are spacious and luxuriously appointed. Bed was comfortable. In-hotel food options were solid** and not as overpriced/marked up as I would have expected".

As the volume of reviews/texts is usually very large, it is necessary to use opinion mining techniques to infer the user's opinion about a given item or about an attribute of the item.

Given the importance of user's texts and the use of opinion mining and contextual information extraction techniques for recommender systems, we present, in this paper, a systematic review on the recommender systems that use information extracted by opinion mining, besides contextual information. This systematic review followed a well-defined protocol. Its results were based on 17 papers, selected among 195 papers identified in four digital libraries. The results of this review give a general summary of the current research on this subject and point out some areas that may be improved in future primary works.

This paper is structured as follows: firstly, we present some concepts of recommender systems and opinion mining in Section 2. In Section 3, we present the systematic review methodology adopted, including the research questions that guided this study and how it was conducted. The obtained results are presented and discussed in Section 4. Finally, in Section 5, we present some final remarks.

2. Background

In this section, we present some important concepts about the two main areas related to this work: recommender systems and opinion mining.

2.1. Recommender Systems

According to Adomavicius and Tuzhilin [10], recommender systems have become an independent area in the mid-1990s and since then these systems have been increasingly used in a number of application fields. Such systems aim to help users by indicating which items they might be interested in, making the user's search easier. The items can be products, services, and people, among others.

Recommender systems can be non-personalized or personalized. Non-personalized systems do not consider the user preferences to make the recommendations. They are based on the most popular items, on the best-evaluated items, and even on newly released items to generate a list of recommendations [21]. Their advantage is that they are simple and can be easily implemented. Their disadvantage is that the items presented to users are always the same, with no recommendations dedicated to the personal interests of each user [21,22]. On the other hand, the personalized recommendation is the task of, based on pre-collected information, estimating users' preferences, or recommending a set of items based on this estimate [1]. In this work, we discuss personalized recommendations.

According to Ricci et al. [1], there are several reasons why service providers may want to explore recommender systems, such as increasing the number of sold items, selling more diversified items, increasing user satisfaction, increasing user loyalty and improving the understanding of user needs and interests.

Recommender systems are information processing systems that put together several data types to build recommendations. Formally, the recommendation problem can be formulated as [10]:

> *"Let U be the set of all users and I the set of all items that can be recommended. Let r be the utility function that measures how useful an item i is for the user u, that is, r : U × I ⟶ R, where R is an ordered set, for example, non-negative integers or real numbers within a given range. Then, for each user u ∈ U, the objective is to find an item i′ ∈ I that maximizes the user's utility, that is, that is more interesting to him:"*

$$\forall u \in U, i'_u = \arg\max_{i \in I} r(u, i). \tag{1}$$

The interest of a user on an item is usually measured by a rating which can be obtained either explicitly or implicitly. In the explicit way, the user tells the system what is his/her opinion on an item (music, item, etc.). According to Schafer et al. [23], explicit ratings can be:

- Numeric: when numerical values are assigned to products/services, for example, the five stars on the Amazon website.
- Ordinal: when the user is prompted to select a term that best indicates his/her opinion on an item, such as "I agree", "I am neutral" and "I disagree".
- Binary: when the user simply decides if an item is good or bad.
- Unary: this kind of ratings was popularized by Facebook where users can mark his/her interest in a post or photo by clicking a button "Like" [24].

In the implicit form, the interests and opinions of the users are collected while they navigate through the site [25]. For example, if a user accesses an item, the system can infer that he/she is somehow interested in that item [1]. Furthermore, the system may consider the amount of time a user spent on a given page to measure his/her interest in it.

The algorithms used by recommender systems can be classified into the following categories [1,26]:

- **Collaborative filtering** [3,27–34]: there are two main methods of collaborative filtering [32], the nearest neighbor methods and the latent factor methods. The nearest neighbor methods are based on the principle that users who have preferred similar items in the past tend to prefer similar items in the future. These methods can be user-based or item-based. In the user-based collaborative filtering, the items (content, services, products, etc.) recommended to a user are those that other users, with similar preferences, have chosen previously. User-based collaborative methods firstly find the users more close to each user, i.e., those with more similar taste and preference. Then, only items that are preferred by these users are recommended to the target user. In the item-based collaborative methods the similarities among different items in the dataset are calculated by using a similarity measure, and then these similarity values are used to predict ratings for user–item pairs not present in the data. The latent factor methods, in turn, are intended to explain users' preferences characterizing users and items by factors, which are characteristics and patterns inferred from existing assessment data. Some of the most successful latent factor algorithms are based on matrix factorization, which characterizes users and items by means of factor vectors. The high match between the factors of items and users leads to a recommendation [35]. Although recommender systems that use collaborative filtering are accurate and efficient, they may present a problem known as cold-start. This problem occurs when the system is unable to make reliable recommendations due to the lack of initial ratings. Another problem faced by collaborative filtering is the sparseness of the data since the number of ratings available is generally very small compared to the number of ratings that need to be provided.
- **Content-based filtering** [36–42]: the items recommended to a user have similar content to the items that this user chose in the past, that is, only the items of high similarity with past user preferences are recommended. Content-based filtering methods have the advantage of not being dependent on the ratings of other users. They may be transparent because explanations about the recommendations are easily generated. In addition, they work well with new items. However, these methods can also present problems, and the two main ones are: (i) limited analysis of content, which is the difficulty in extracting reliable information automatically from various types of content such as images, videos, audios and texts; and (ii) super-specialization, as the system recommends items by analyzing the user's profile, this causes items to be very similar to items that were previously accessed by the user.
- **Hybrid approaches** [43–51]: these approaches aim to benefit from the advantages of each type of approach, reducing the problems that they present. Thus, the hybrid approaches combine collaborative and content-based methods. The combination can be done in some ways. For example, one type of combination is to implement such methods separately and combine the results to produce the final recommendations. Another way is to use both in a single recommendation model.

Although recommender systems using collaborative filtering techniques are accurate, they can present a problem known as "cold-start". This problem occurs when the system can not make reliable recommendations due to the lack of initial ratings or necessary information [9]. Another problem faced by these systems is the sparsity because the number of available ratings is usually very small compared to the number of ratings that need to be provided.

Content-based filtering methods also have some problems, and the two main ones are [1,9]: (i) limited analysis of the contents, which refers to the difficulty in extracting reliable information automatically from various content such as images, video, audio and text; and (ii) super-specialization, as the system recommends items analyzing the user profile, the user is restricted to see similar items to those that he/she has already assessed/seen before.

It is possible to note that hybrid approaches, as well as collaborative filtering and content-based approaches, when in the traditional forms, focus only on the entities' items and users to build the recommendation model [1,9]. This process of recommendation is known as two-dimensional because it considers only User × Item dimensions to generate the recommendations. However, in many applications, it is also important to incorporate contextual information into the recommendation process [17,52,53]. For example, a travel package recommended in summer can be different from the package recommended in winter; a person may prefer to read economics and political news during the week, but, on the weekend, he/she may want to read news about sports or celebrities; the film indicated for a person may depend on the time of day, perhaps in the evening, the preference is horror movies, while, during the day, the preference is comedy.

Context-aware recommender systems make recommendations by considering contextual information. The importance of contextual information has been recognized by researchers and professionals in many areas, such as personalization of e-commerce websites, information retrieval and mobile computing [54]. Thus, several applications can be automatically customized to improve interaction with their users [55–59]. The context-aware recommender systems model and predict the tastes and preferences of users by incorporating available contextual information in the recommendation process. According to Adomavicius and Tuzhilin [54], tastes and preferences of users are generally expressed as ratings and modeled on the basis of items, users and contextual information.

Contextual information is a concept that can have different definitions depending on the area in which it appears. The most widely used definition was suggested by Dey [18]: "Context is any information that can be used to characterize the situation of an entity. An entity can be a person, a place, or an object that is considered relevant to the interaction between a user and an application, including the user himself/herself and the applications themselves".

According to Adomavicius and Tuzhilin [54], contextual information can be applied at various stages of the recommendation process; and, according to this criterion, the systems can be divided into three categories, as shown in Figure 1: (i) contextual pre-filter, (ii) contextual modeling, and (iii) contextual post-filtering.

In the contextual pre-filter approach, the contextual information is used to select the dataset that will be used for the learning of the recommendation models. The recommendations can be made by using a traditional recommender system and taking as input the selected contextual data. An advantage of this approach is that it allows the use of some traditional recommender system already proposed. For example, if a person wants to watch a movie on Saturday, we can generate a set of recommendations for him/her by applying a traditional recommendation approach with the ratings made on Saturdays as contextual data input [54].
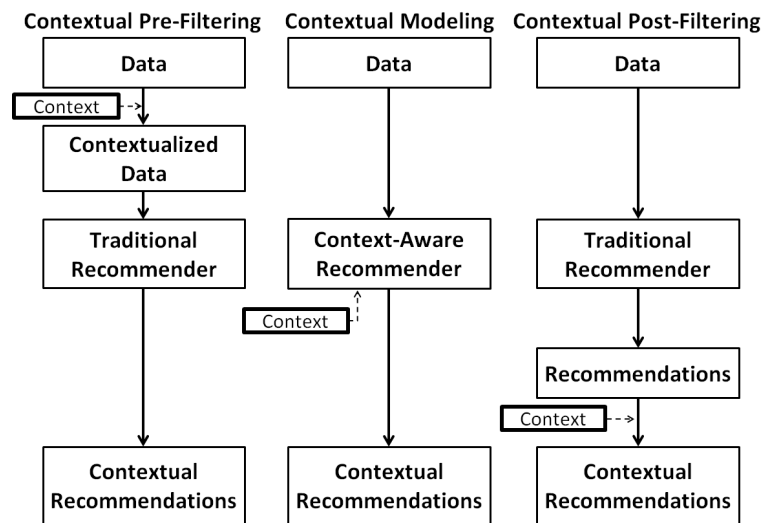
Figure 1. How to use context in the recommendation process (adapted from Panniello and Gorgoglione [60]).

Adomavicius et al. [52] proposed a reduction based approach which reduces the multidimensional space of the context-aware recommender systems in a traditional two-dimensional space *User × Item*:

"Let $R^D_{User \times Item} : U \times I \longrightarrow Ratings$ a function to estimate ratings that, given the existed ratings D, can calculate a prediction for any rating. Then, a three-dimensional rating prediction function that accepts the time information (contextual information) can be defined as $R^D_{User \times Item \times Time} : U \times I \times T \longrightarrow Ratings$. It can be expressed by a two-dimensional prediction function in various ways, and one of these ways is:

$$\forall (u, i, t) \in U \times I \times T, R^D_{User \times Item \times Time}(u, i, t) = R^{D[Time=t](User, Item, Rating)}_{User \times Item}(u, i),$$

where $[Time = t]$ denotes a simple contextual pre-filter, and $D[Time = t](User, Item, Rating)$ denotes a set of ratings D selecting only the records in which the dimension T has value t and keeping only the values for the dimensions *User* and *Item*, as well as the rating value itself".

In the contextual post-filtering approach, the contextual information is used after the construction of a traditional recommendation model to filter or reorder the recommendations. First, the top-N recommendations are generated, and then the contextual post-filtering approach adjusts the list of recommendations obtained for each user using the contextual information. According to Adomavicius and Tuzhilin [54], the adjustments to the list of recommendations can be made by:

- Filtering out the recommendations that are irrelevant in a given context; or
- Adjusting the ranking of the recommendations in the list based on a certain context.

For example, if a person wants to see a movie on Sunday and on Sundays he/she only watches horror movies, then the system can only consider the recommendations of horror movies to show to the user.

Panniello and Gorgoglione [60] proposed the contextual post-filtering approaches known as *Weight PoF* and *Filter PoF*. Both analyze the data for a given user in a specific context by calculating the probability of the user to choose a particular item in this context. After that, the recommendations obtained by using a traditional approach are contextualized with the probabilities calculated.

The contextual probability $P_C(u, i)$, in which the user u accesses the item i in the context C, is calculated as the number of neighbors (similar users to u) that accessed the same item in the same context, divided by the total number of neighbors. The *Weight PoF* and *Filter PoF* approaches differ in the way that the recommendations are contextualized. In particular, in the *Weight PoF* approach each rating is multiplied by the probability $P_C(u, i)$:

$$Rating_C(u, i) = Rating(u, i) \times P_C(u, i),\tag{2}$$

whereas the *Filter PoF* approach filters out the ratings by using a threshold value P*:

$$Rating_C(u, i) = \begin{cases} Rating(u, i), & \text{if } P_C(u, i) \geq \text{P*}, \\ 0, & \text{if } P_C(u, i) < \text{P*}. \end{cases}\tag{3}$$

In the contextual modeling approach, the context is used in the recommendation models, i.e., the contextual information is part of the model along with the user and items data. Generally, true multidimensional recommendation functions are used. These functions can represent predictive models such as decision trees, regression, and probabilistic models, or may represent heuristic calculations that incorporate contextual information.

Domingues et al. [53] proposed an approach that uses the contextual attribute as a virtual item, that is, this attribute is treated as a common item to build the recommendation model. Thus, this approach called *DaVI (Dimensions as Virtual Item)* also allows the use of traditional recommendation algorithms.

Formally, let $U = \{u_1, u_2, \ldots, u_m\}$ be the user set and $I = \{i_1, i_2, \ldots, i_n\}$ be the item set, there are other dimensions, for example, contextual information, $D = \{D_1, D_2, \ldots, D_t\}$, where each dimension D comprises a set of values, i.e., $D = \{d_1, d_2, \ldots, d_f\}$. Let j be the number of multidimensional sessions $S' = \{s'_1, s'_2, \ldots, s'_j\}$. Each session s' is a tuple defined by a user $u \in U$, an accessed item set $I_{s'} \subset I$ and a set $D_{s'} \subseteq D_1 \cup D_2 \cup \ldots \cup D_t$ containing all the dimension values associated with the session s', i.e., $s' = \langle u, I_{s'}, D_{s'} \rangle$.

The *DaVI* approach consists in converting each multidimensional session $s' = \langle u, I_{s'}, D_{s'} \rangle$ in an extended two-dimensional session $s'' = \langle u, I_{s''} \cup D_{s''} \rangle$, in which the additional dimension values $D_{s''}$ are used as virtual items along with the actual items from I_s''.

It is important to determine which dimensions must be included in a recommendation model, as some dimensions are more informative than others. Domingues et al. [53] proposed the algorithm *DaVI-BEST* that evaluates and selects the best dimension of a set of data to build the multidimensional recommendation model. To determine the best dimension for a given recommendation algorithm A, the *DaVI-BEST* algorithm first applies the *DaVI* approach in each candidate dimension and builds its multidimensional recommendation model. The approach then evaluates the model and selects the best dimension, one whose recommendation model presents the best performance. In this algorithm, the *F1 measure* is used to evaluate each recommendation model.

The main gap in the area of context-aware recommender systems is the lack of automatic methods for the contextual information acquisition. Many websites allow users to provide reviews about services and products offered in the domain. For example, in the IMDb website [61], users write reviews commenting on different characteristics of the films (actors, effects, etc). Such opinions have a major impact on consumer decisions since users rely more on reviews provided by users than on those obtained by other means, including automatic recommenders. Thus, reviews are valuable data generated by users, since, in general, they, besides evaluating items, also explain why they liked or did not like them and give indications of contextual information. The collections and syntheses of these opinions are carried out by opinion mining systems. In the next section, we introduce the opinion mining area.

2.2. Opinion Mining

With the growth of social networks, more and more users can openly discuss their impressions and experiences on a variety of products, items, and services. This means a significant increase in user-generated content in the form of reviews, blogs, discussion forums, social networks, etc. Among this content, reviews represent rich sources of data and they are very useful for marketing intelligence, social psychology and other areas that are interested in mining opinions, views, sentiments and attitudes [62].

The reviews are available in the Web in several websites, as websites dedicated to specific products, websites of newspapers and magazines, websites of e-commerce and websites specialized in collecting reviews from customers or professionals from various fields. A very important type of information found in such content is the opinion. The opinions are the center of almost all human activities and are an influencer of behaviors [63]. When people need to make a decision, they often look up for the opinions of others. An opinion of a user about a product, service or subject typically reveals his/her satisfaction or dissatisfaction on it and how much he/she cares about certain specific characteristics of the item. For this reason, online reviews are very useful when deciding to buy a product, see a movie or go to a restaurant. Moreover, companies receive feedback from users through reviews [64].

Businesses and individuals are increasingly using the content of reviews to make better decisions. However, reviews are usually written in a free text format, making it difficult for basic computer systems to interpret, analyze and aggregate them. In this way, opinion mining systems are used to collect and display the opinions.

According to Liu [63], opinion mining is the field of study that examines people's opinions, feelings, assessments, attitudes, and emotions related to entities and their aspects (features/components/attributes). In other words, opinion mining aims to infer the review author's attitude in relation to a particular item. The opinion can be expressed in a certain time and context and in relation to the object as a whole or to any of its aspects, including aspects of the aspects. It is important to know that an entity is a product, service, topic, subject, person, organization, or event, and the aspects are the parts and attributes of the entity.

The term "opinion mining" first appeared in [65]. However, research on sentiments and opinions emerged earlier [66,67]. Opinion mining usually implies the use of text analysis techniques, natural language processing and computational linguistics to identify, extract and understand subjective content (which has an opinion).

There are some key concepts and definitions in the field of opinion mining, such as [63]:

- **Opinion**: Liu [63] quotes the definition of opinion from the Merriam-Webster dictionary in which opinion is "a view, judgment, or evaluation formed in the mind about a specific subject". In this work, the opinion includes feeling, evaluation, attitude and associated information, as the target of the opinion and the person who holds the opinion.
- **Sentiment**: following the definition of the same dictionary (*Merriam-Webster*), sentiment is "an attitude, thought, or judgment caused by perception". Liu [63] draws the attention of the readers of his book to the great similarity between definitions of opinion and sentiment but concludes that opinion is better defined as a concrete view of a person about something, whereas sentiment is a perception. Sentiment is considered as a positive, negative, and sometimes neutral perception about a particular subject.
- **Document (*h*)**: it is a natural language text that reports on a particular subject, theme, problem, product, organization, among others.
- **Document set (*H*)**: it is a set of documents about one or more specific subjects.
- **Entity (*e*)**: it is a product, topic, service, person, organization or event that is being referenced in the documents. An entity is described by a set of components and their aspects [63]. Entities can be mentioned in some works as objects.
- **Aspect (*a*)**: it is a property, component or feature of an entity. Examples of aspects are product size, product price, service quality, and so on. In the literature, aspects can be termed as features or attributes.

There are different levels of opinion mining [63]: (i) document level, in which the task is to classify the opinion document in positive or negative, i.e., identifies the global sentiment of the document; (ii) sentence level, level at which the sentiment polarity expressed in each sentence is identified (positive or negative); and (iii) entity and aspect level, in this level, the entities and aspects are extracted and the opinion about each of them is classified in positive or negative. Previous levels may fail to find what the opinion's author likes or dislikes. A positive document on an entity does not mean that

the opinion's author will have positive opinions on all aspects of that entity. In addition, a negative document does not mean that the opinion's author does not like anything about the entity, i.e., both previous levels do not discover what exactly people like or not. For example: "The iPhone call quality is good, but the battery is short-lived". In this example, from Liu [68], the entity is the "iPhone" and the opinions are expressed on two aspects: call and battery life. The call is evaluated with a positive feeling and battery life with a negative feeling.

According to Liu [63], an opinion is a quintuple: $(e_i, a_{ij}, s_{ijkl}, g_k, t_l)$, where e_i is the name of an entity, a_{ij} is an aspect of e_i, s_{ijkl} is the sentiment for the aspect a_{ij} of e_i, g_k is the opinion author, and t_l is the time when the opinion was expressed. The sentiment s_{ijkl} can be positive, negative or neutral, or may be expressed at different intensity levels, for example from 1 to 5 stars, as used for several sites. The purpose of opinion mining is discovering all the quintuples given an opinion document h.

Aspect-based opinion mining consists of a complex process involving techniques from various research areas (for example, natural language processing, data mining, machine learning, linguistics, and even social science). This process is described below and related to the example of Figure 2.

1. **Entity Extraction and Categorization**: identify all entity expressions in h and categorize the synonyms in entity categories e_i.

    In the example: the expressions "Samsung", "Samy" and "Canon" are identified, being that the first two represent the same entity "Samsung Camera".

2. **Extraction and Categorization of Aspects**: identify all entity feature expressions and categorize these expressions into categories ($a_{i,j}$).

    In the example: the expressions "image", "photo" and "battery life" are identified, being the first two the representation of the same aspect "image".

3. **Identification of the opinion holder**: identify who issued the opinion.

    In the example: in sentence (3), the opinion author can be bigJohn and in the sentence (4) can be the friend of bigJohn.

4. **Extraction and Time Standardization**: to identify when opinions have been published and to standardize the different time formats.

    In the example: the message was posted on 15 September 2011. A default format could be 2001-09-15.

5. **Classification of Aspect Sentiments**: to determine the polarity of the sentiment on an aspect $a_{i,j}$, that is, to classify the sentiment as positive, negative or neutral.

    In the example: sentence (3) gives a negative opinion of the image quality and battery life of Samsung camera. Sentence (4) gives a positive opinion to the camera as a whole and also to its image quality. To generate the opinion quintuples contained in the sentence (4), it is necessary to know to which camera the expressions refer to: "his camera" and "his".

6. **Generation of Opinion Quintuples**: to generate all opinion quintuples $O = (e_i, a_{ij}, s_{ijkl}, g_k, t_l)$ expressed in the documents of the collection.

    In the example: O_1 = (Samsung, image_quality, negative, bigJohn, 2001-09-15)

    In the example: O_2 = (Canon, overall, positive, bigJohn's friend, 2001-09-15)

7. **Summarization**: opinions are ordered, categorized and summarized so that the entities, their aspects and their sentiments about the target object are presented, allowing a better interpretation of the data.

> Posted by: bigJohn   Date:15/set/2011
>
> "(1) I bought a Samsung camera and my friend bought a Canon camera yesterday. (2) In the last week, we used the cameras very much. (3) The photos from my Samy are not so great, and the battery life is too short. (4) My friend was too happy with his camera and he loved the image quality. (5) I want a camera that can take good pictures. (6) I will give it back tomorrow."

Figure 2. Example of opinion text or review (Adapted from Liu [68]).

According to Liu [63], there are four types of approaches for identifying aspects:

1. **Frequency-based**: an aspect can be expressed by a noun, adjective, verb or adverb, but studies show that from 60 to 70% of explicit aspects are nouns [69]. Aspects tend to be frequent nouns since, in commentaries, people are generally more likely to talk about the relevant aspects. However, there are nouns that are not aspects and aspects that are not nouns. In this way, different selection techniques are applied to frequent nouns to identify which of these are aspects. In general, frequency-based methods generate a set of candidate aspects and use a selection criterion that can be based on co-occurrence, syntactic pattern, Point-wise Mutual Information (PMI) measure, among others [70–72].
2. **Based on syntactic relations**: there are usually many syntactic relationships between the expressions of sentiment and the opinion targets. Such relationships are possible to be explored when words and phrases of sentiment are known. If the sentence does not have a frequent aspect but has some words of sentiment, the noun closest to a sentiment word is extracted as an aspect [70,73].
3. **Through supervised learning**: in general, methods for identifying aspects are based on *sequential labeling*. The most commonly used methods are: Conditional Random Field (CRF) [74] and Hidden Markov Model (HMM) [75].
4. **Through topic models**: topic modeling is an unsupervised learning method that assumes that each document is composed of a set of topics and each topic has a probability distribution over words. The main topic models used for the extraction of aspects are: Latent Dirichlet Allocation (LDA) [76] and Probabilistic Latent Semantic Indexing (PLSI) [77].

For the task of classifying sentiments of the aspects, there are two approaches [63]:

1. **Based on machine learning**: for classification of sentiments related to aspects, traditional machine learning algorithms such as SVM and Naive Bayes, which are used for the classification of sentiments at the sentence and document levels, are not enough [63]. The main reason is that these algorithms do not consider an opinion target (entity and/or aspect) and therefore are unable to determine what the classified sentiment refers to. To solve this problem, it is necessary to adapt the algorithms so that they are able to consider a target of opinion in the learning process. To do this, the main current approach is to use parsing to determine dependency and other pertinent information.
2. **Based on lexicon**: lexical-based classifiers are generally unsupervised. In general, a lexical approach to rating sentiments about aspects uses the following features [78,79]:

   - a lexicon of sentiment expressions including words of sentiment, phrases, idiom expressions and rules of composition;
   - a set of rules for dealing with different language constructs (for example, sentiment modifiers and but-clauses) and types of sentences; and,
   - a sentiment aggregation function or a set of sentiment and target relationships derived from the syntactic tree to determine the orientation of the sentiment at each destination in a sentence.

Thus, the opinion mining process may extract relevant information about the items. Given the importance of aggregating additional information in recommender systems, such as contextual and opinion information, we investigated works of the literature that use opinion mining along with context-aware recommender systems. The purpose of this research was to answer the research questions presented in Section 3.1. For this, we conducted a systematic review presented in the next section.

3. Systematic Review

Conducting a systematic review means to conduct bibliographic reviews in a formal way by following a well-defined protocol. This kind of literature review aims to specify questions and to review relevant studies, which allow researchers to identify gaps in the current research, as well as to propose their contributions to the area. Because of the well-defined steps of the systematic review's protocol, the results can be reproducible.

Our systematic review was conducted based on the methodology introduced by Kitchenham and Charters [80], which involves three major phases: planning, conducting and presenting and discussing the results. In the planning phase, the needs of the systematic review and the relevant bibliographic datasets are identified, the research questions are specified and the search expression, the selection criteria, and the information extraction strategy are defined. In the conducting phase, the search expression is applied and the returned papers are selected according to the criteria defined in the previous phase. Then, the selected papers are read and the relevant information is extracted. Finally, in the last phase, the results are presented and discussed. Each phase is detailed in the following sections.

3.1. Planning the Review

During the planning phase, we conducted the following activities:

- **Identification of the need for a review**—in the first phase, we identified that there is no systematic review in the field of context-aware recommender systems that use opinion mining. Some systematic reviews were published in the recommender system area, with or without contextual information, like [81–84], but none of them considered opinion mining and context together. Thus, identifying and analyzing the works that consider contextual information and opinion mining in recommender systems would be of great help to the research community.
- **Specification of the research questions**—we specified the following research questions:

  1. What contextual information has been adopted for making recommendations?
  2. How has the contextual information been extracted?
  3. What opinion information has been adopted for making recommendations?
  4. How has the opinion information been extracted?
  5. Which textual sources have been used for the extraction of both context and opinion information?

- **Identification of the relevant bibliographic datasets**—in order to find the relevant studies for the review, we chose the bibliographic datasets that cover the majority of journals and conference papers published in the field of computer science. The selected bibliographic datasets were: Scopus [85], ACM Digital Library [86], IEEE Xplore Digital Library [87], and ScienceDirect [88].
- **Definition of the search expression**—after defining the research question, we built the search expression. The used search expression underwent some changes as coverage issues were observed, that is, when the search was too broad or too restrictive. The final version of the search expression is: `(context*) AND ((recommender system* OR recommendation system*)) AND ((sentiment*) OR (opinion*))`.
- **Definition of the selection criteria**—in this step, we defined the selection criteria, that is, the criteria used to include or exclude the papers. Every paper returned in the search phase went to the selection phase. In the selection phase, we eliminated duplicate papers and analyzed the remaining studies in order to exclude the ones that match at least one of the following exclusion criteria.

- Secondary studies, i.e., reviews or surveys.
- Publications that do not deal with context-aware recommender systems that use opinion mining. Therefore, the works about recommender systems that consider only contextual information or only opinion mining were not included.
- Publications with one page, posters, presentations, abstracts, and editorials.
- Publications hosted in services with restricted access and not accessible or publications not written in English.

The reading of the papers was performed in the following order: (i) title, abstract, and keywords; (ii) introduction and conclusion; and (iii) full paper.

- **Definition of the information extraction strategy**—in order to collect the information needed to answer our research questions, our information extraction strategy was defined as to read the full-text of every paper that was accepted in the selection phase (papers that were not identified as duplicated or rejected). We defined the following information to be extracted from each selected paper. Numbered lists can be added as follows:

1. **Bibliography data:** title, authors, publication year, journal or conference.
2. **Study data:** adopted contextual information, method used for contextual information extraction, opinion information adopted in recommendations, method used for opinion information extraction, textual sources of contextual and opinion information, domain of the recommender system, and opinion mining level.
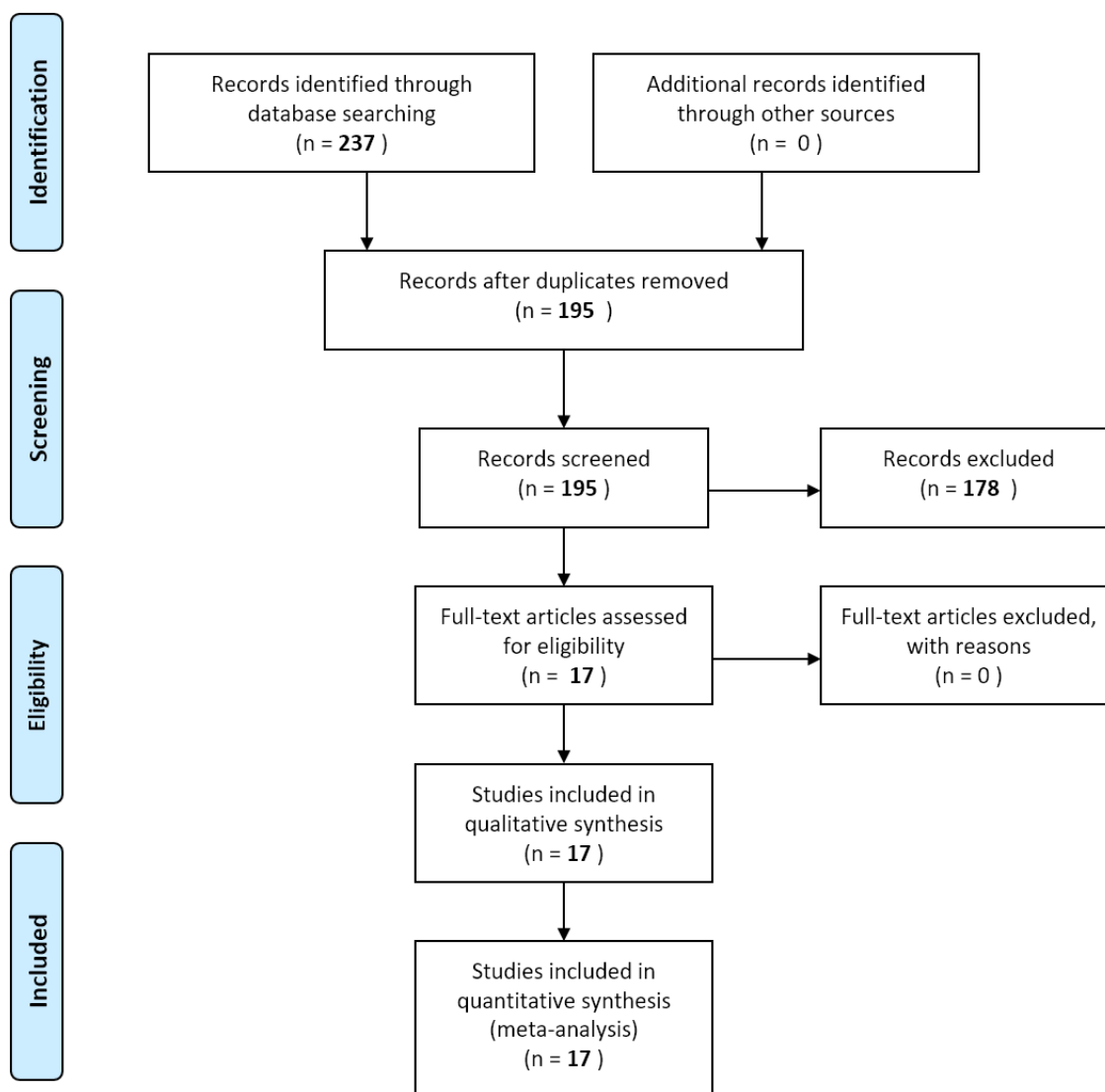
3.2. Conducting the Review

The conduction of this systematic review followed the defined protocol, whose main parts were presented in the last subsection. The PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) Flow Diagram [89] of our systematic review (Figure 3) presents the information flow through its phases, mapping the number of papers identified, included and excluded during the systematic review phases.

The first step of the systematic review conduction (Identification phase) was the application of the search expression in each bibliographic dataset. These searches were executed in 10 October 2018, and they resulted in 160 returned papers from Scopus, 32 from IEEE Xplore Digital Library, 26 from ACM Digital Library, and 19 from ScienceDirect. Thus, a total of 237 papers were found using the search expression in the four digital libraries.

In the Screening phase, after the removal of 42 duplicate studies, the 195 remaining papers were assessed based on their title, abstract, keywords, introduction, and conclusion. In this phase, 178 papers were excluded according to the exclusion criteria presented in Section 3.1. Among the excluded papers, only one was excluded by the secondary study criteria. The work of Chen et al. [20] is a review of the literature on recommender systems based on user reviews. Although it is an important review on the subject, it is focused on user reviews, whereas our study is focused on opinion mining.

After the Screening phase, 17 papers were included in the information extraction phase. In this phase, the 17 papers were read and the information about each study was extracted. The results of this systematic review are presented in the next section.

Figure 3. Information flow through the phases of the systematic review. PRISMA Flow Diagram adapted from Moher et al. [89].

4. Results and Discussion

The review reported in this paper was conducted with the general goals of: (i) identifying the research on context-aware recommender systems that also use information extracted by opinion mining; (ii) and mapping how this research is combining these two technologies (context-aware recommender systems and opinion mining). The results of this systematic review are presented in this section. Firstly, in Section 4.1, we present an overview of the research fields and the selected papers, and we also answer the research questions. Detailed descriptions of the selected papers are presented in Section 4.2.

4.1. Selected Studies and Research Questions

Research on recommender systems that use both context and opinion information involve two main research fields: context-aware recommender systems and opinion mining. To illustrate the evolution of these fields, Figure 4 presents the number of published documents over the last 20 years. The data of Figure 4 was obtained through searching through Scopus' Computer Science subject area (The searches were performed on 12 December 2018 and the applied search strings were: `TITLE-ABS-KEY((context* AND ("recommender system*" OR "recommendation system*")))`

AND (LIMIT-TO(SUBJAREA,"COMP")) and TITLE-ABS-KEY((("opinion mining") OR ("sentiment analysis"))) AND (LIMIT-TO(SUBJAREA,"COMP"))). We can note that both areas have become more active over the last ten years, with an increasing number of publications.
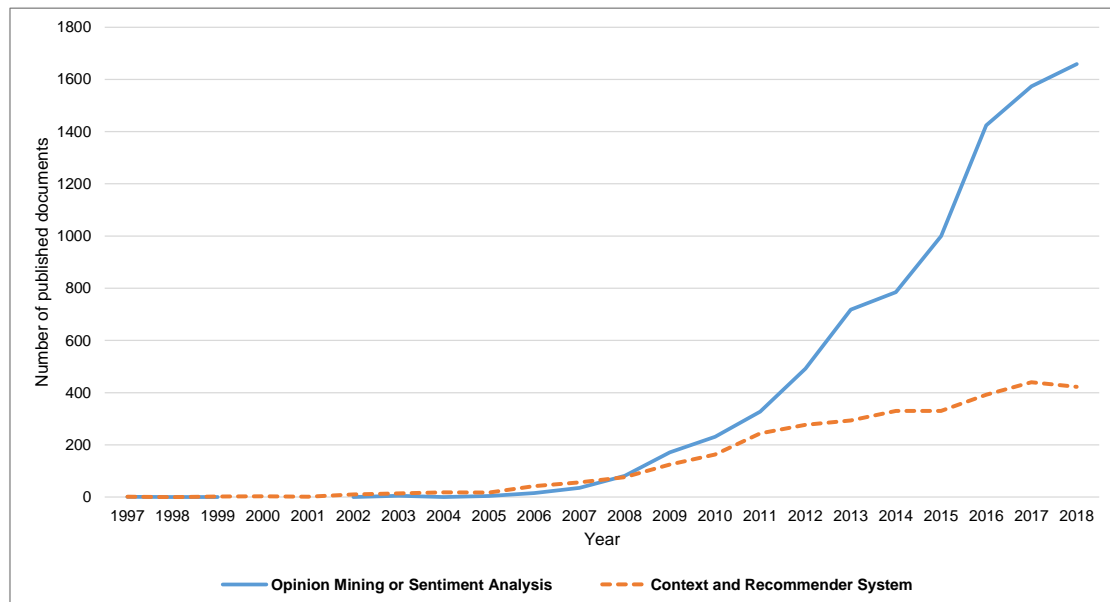
Figure 4. Evolution of research fields related to the systematic review.

Following the defined protocol (Section 3), we identified and reviewed 17 papers that report research on recommender systems that use both context and opinion information. These papers were published between 2012 and 2018. In Table 1, we summarize these 17 selected works that use context-aware recommender systems and opinion mining. In some fields of the table, the value "not mentioned" is observed, which means that it was not possible to extract the information of a certain column for the corresponding work. The columns of the table are:

1. **References**: references of the studies on context-aware recommender systems that use opinion mining.
2. **Domain**: domain of the recommender system addressed in each job, for example, "hotels", "tourism", among others.
3. **Contextual information**: column consisting of three sub-columns referring to the contextual information used in each system:

   - **Type**: the kind of contextual information, which can be "location", "time", "occasion", etc.
   - **Automatic extraction**: "yes", if the contextual information is extracted automatically, that is, it does not need to be informed by the user; and "no" otherwise.
   - **Predefined values**: "yes", when it is necessary to define the values of the contextual information to search for such values, (for example, string matching); and "no" otherwise.

4. **Opinion mining**: column formed by two sub-columns referring to the opinion mining executed in each work:

   - **Aspect level**: "yes", if the opinion mining performed at work is at the level of aspects; and "no" otherwise.
   - **Predefined aspect values:** "yes", if aspect values need to be predefined to be search (for example, string matching); "no", when values are not predefined; and "does not apply" when opinion mining is not at the level of aspects.

Table 1. Summary of works on context-aware recommender systems using opinion mining.

| Reference | Domain | Contextual Information | | | Opinion Mining | |
		Type	Automatic Extraction	Predefined Values	Aspect Level	Predefined Aspect Values
Ho et al. [90]	spatio-temporal events	location and time	yes	no	no	does not apply
Levi et al. [91]	hotels	travel type and nationality	no	yes	yes	no
Meehan et al. [92]	tourism	location, weather, time, sentiment and personalization	yes	no	not mentioned	not mentioned
Chen and Chen [93]	restaurants	time, occasion and companion	yes	yes	yes	yes
Chen and Chen [19]	restaurants and hotels	time, occasion and companion	yes	yes	yes	yes
Colace et al. [94,95]	general	item features from search or location	no	no	no	does not apply
Kothari and Patel [96]	not mentioned	not mentioned	based on [93]	based on [93]	based on [93]	based on [93]
Orellana et al. [97]	short films	affective context (emotion)	yes	yes	no	does not apply
Yang et al. [98]	places	location	not treated	not treated	no	does not apply
Zhao et al. [99]	restaurants, nightlife and movies	location and time	not mentioned	no	no	does not apply

Table 1. *Cont.*

| Reference	Domain	Contextual Information			Opinion Mining	
		Type	**Automatic Extraction**	**Predefined Values**	**Aspect Level**	**Predefined Aspect Values**
Kharrat et al. [100]	movies	users' opinions for tags of items	yes	no	yes	yes
Missaoui et al. [101]	tourism	location	yes	no	no	does not apply
Jalan and Gawande [102]	hotels	travel type location opinion	no not mentioned yes	yes not mentioned no	yes	no
Baral et al. [103]	places	category location opinion	not mentioned not mentioned yes	not mentioned not mentioned no	yes	no
Sulthana and Ramasamy [104]	books	opinion	yes	no	yes	no
Zangerle [105]	music	emotional state (emotion)	yes	no	no	does not apply

Based on the 17 selected papers, we answer the research questions in the following. Questions 1 (*"What contextual information has been adopted for making recommendations?"*) and 2 (*"How has the contextual information been extracted?"*) are related to the contextual information. In Table 2, we answer these research questions according to each work. We can note that location and time are the most frequent contextual information used by the reviewed recommender systems. Location is used as context information in seven research studies and time is used in four of them. Another interesting finding is that both opinion and emotion appear as contextual information in two works each. Considering the extraction method, we can find studies obtaining contextual information from: (i) external systems, such as GPS and *WorldWeatherOnline API*; (ii) user profile or user specification; and (iii) matching texts with predefined values or lexical resources.

Questions 3 (*"What opinion information has been adopted for making recommendations?"*) and 4 (*"How has the opinion information been extracted?"*) are related to opinion information. In Table 3, we answer these research questions according to each reviewed paper. Fourteen (14) of the 17 research studies uses sentiment as the opinion information. However, only seven works apply opinion mining at the aspect level (see Table 1). The methods used for opinion information extraction are varied, including supervised and unsupervised methods and the use of lexical resources.

Our fifth research question is *"Which textual sources have been used for the extraction of both context and opinion information?"*. We answer this research question according to each reviewed paper in Table 4. Although there are several available sources of user-generated content, reviews are still the main source used by in context-aware recommender system solutions.

We also present an overview of the 17 selected papers through a relationship graph (Figure 5), where each node is a paper and the edges indicate significant similarity between papers. This analysis is important to identify common topics among the papers. To compute the similarity, we represent each paper with features extracted from the title, abstract, keywords, and type of contextual information. We use the cosine similarity and term frequency - inverse document frequency (TF-IDF) weighting, which are suitable techniques for textual data analysis. Moreover, we apply a graph clustering technique to identify groups of related papers. The most relevant features of each group were used to label common topics among papers.
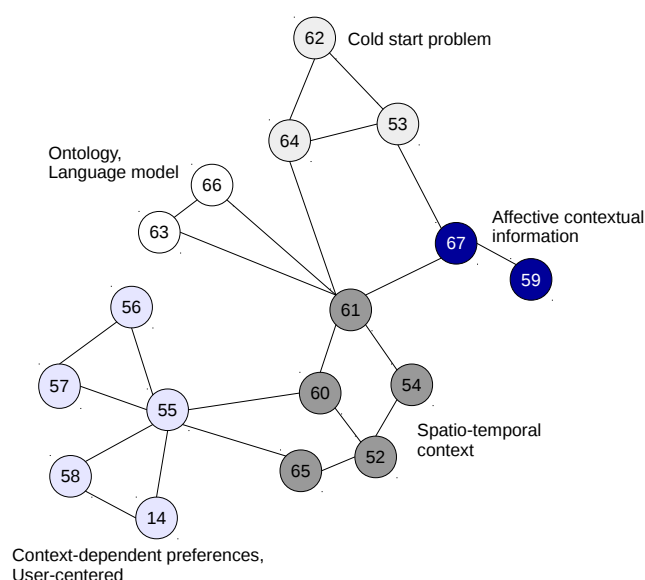
Figure 5. Overview of the similarity relationship graph between papers selected in the systematic review and its organization in groups of topics. Each paper is a graph node and the node number indicates the bibliographic reference number. Two similar papers are connected by an edge. Colors define groups of papers.

Table 2. Answers for research questions related to contextual information (questions 1 and 2).

Reference	Contextual Information	How Contextual Information Is Extracted
Ho et al. [90]	• location • time	• location: toponym recognition and resolution using POS tagging, named entity recognition and rule-based heuristic recognition • time: recognition of feature and near past temporal patterns
Levi et al. [91]	• nationality • travel type	defined by users in reviews
Meehan et al. [92]	• location • time • weather • sentiment • personalization	• location: GPS, GSM and Wi-Fi • time: the amount of time that the person stays at the place is considered to calculate the interest level in this place • weather: *WorldWeatherOnline API* • personalization: data like age, gender, relationship and number of children are extracted from user profile
Chen and Chen [19,93]	• time • occasion • companion	types and values of the contextual information are manually defined and a string matching method is applied
Colace et al. [94,95]	• item features from search or user position	informed by user
Kothari and Patel [96]	not mentioned	based on [93]
Orellana et al. [97]	• affective context (emotion)	*Automatic Emotion eXtraction* uses *LingPipe* and *MorphAdorner POS* tools and *EmoLex*
Yang et al. [98]	• location	the contextual information is not extracted
Zhao et al. [99]	• location • time	not mentioned
Kharrat et al. [100]	• users' opinions for tags of items	extracts item tags using predefined tags and opinion tags using *WordNet*
Missaoui et al. [101]	• location	using GPS
Jalan and Gawande [102]	• travel type • location	not mentioned
Baral et al. [103]	• category • location	not mentioned
Sulthana and Ramasamy [104]	• opinion	nouns are extracted as aspect terms and sentiment polarities are extracted using the *SentiWordNet*
Zangerle et al. [105]	• emotional state (emotion)	hashtags containing user emotion state are considered and matched with sentiment lexica

Table 3. Answers for research questions related to opinion information (questions 3 and 4).

Reference	Opinion Information	How Opinion Information Is Extracted
Ho et al. [90]	sentiment of the event	using two classification appraches: sLDA and SVM
Levi et al. [91]	sentiment of item aspects	an unsupervised community detection technique based on [106] is used to extract the aspects and a bootstrapping lexicon-based approach based on [70] is used to extract the aspect sentiments
Meehan et al. [92]	sentiment of the tourist points	using AlchemyAPI
Chen and Chen [19,93]	sentiment of item aspects	using the bootstrapping method proposed in [107] to identify the aspects and a opinion lexicon to detect the aspect sentiments
Colace et al. [94,95]	sentiment of reviews	using an improvement of the approach presented in [108], where the LDA is applied
Kothari and Patel [96]	based on [93]	based on [93]
Orellana et al. [97]	affective context (emotion)	*Automatic Emotion eXtraction* uses *LingPipe* and *MorphAdorner POS* tools and *EmoLex*
Yang et al. [98]	sentiment of reviews	using the ratings
Zhao et al. [99]	sentiment	using the sentiment analysis method proposed in [109]
Kharrat et al. [100]	opinion words	using *WordNet*
Missaoui et al. [101]	sentiment of reviews	using the ratings
Jalan and Gawande [102]	sentiment of item aspects	nouns are extracted as aspect terms and are grouped as aspects. The aspect sentiment is extracted using a lexico and considering the adjectives surrounding the aspect
Baral et al. [103]	sentiment of item aspects	aspect terms are extracted using a noun frequency approach. The terms are categorized in aspects using *WordNet*. The sentiments are extracted using the trigram around the aspect terms
Sulthana and Ramasamy [104]	sentiment of item aspects	nouns are extracted as aspect terms and sentiment polarities are extracted using the *SentiWordNet*
Zangerle et al. [105]	emotional state (emotion)	hashtags containing user emotion state are considered and matched with sentiment lexica

Table 4. Answers for the research questions related to textual sources of contextual and opinion information (question 5).

Textual Source of Contextual or Opinion Information	References
news articles	• Ho et al. [90]
reviews	• Levi et al. [91] • Chen and Chen [19,93] • Colace et al. [94,95] • Kothari and Patel [96] • Yang et al. [98] • Zhao et al. [99] • Kharrat et al. [100] • Missaoui et al. [101] • Jalan and Gawande [102] • Baral et al. [103] • Sulthana and Ramasamy [104]
mobile applications	• Meehan et al. [92]
tweets	• Meehan et al. [92] • Zangerle et al. [105]
user comments	• Orellana et al. [97] • Kharrat et al. [100]

We emphasize that, in addition to the organization by type of contextual information and the use of opinion mining described in Tables 1–3, the graph analysis complements the visualization of the revised works by showing the main characteristics in common from a statistical point of view. The group with papers [90,92,98,99,103] has, as a main characteristic, the use of location and time as contextual information, while the group with papers [97,105] more directly explores emotion and affective information as a context in the recommendation system. The group with papers [19,93–96] focuses on analyzing user's preference extracted from opinion mining and on the use of this information as context-dependent preferences. The group with papers [91,100,102] has as a differential to deal with the cold start problem in context-aware recommendation systems by using opinion mining to obtain an initial model when there is not enough information for collaborative filtering. Finally, the group with papers [101,104] explores ontologies and language models to support opinion mining in the context-aware recommendation systems. Note that, even when papers are allocated in different groups, there is still a relationship between them. For example, papers [102] and [99] are allocated in different groups but are connected in the graph by sharing some type of context information (e.g., location).

As previously stated, this systematic review considered studies in the intersection of two research topics. "Context and Recommender Systems" and "Opinion Mining and Sentiment Analysis" are recent research topics, with an increasing number of publications in Computer Science (Figure 4). Based on the results presented in this section, we found that the number of research works that effectively uses opinion mining in context-aware recommender systems is still low (17 primary studies). Although those selected works are related to the same main topic, each group emphasizes different aspects of the broad theme (Figure 5). They are very relevant and present important advances in the use of opinion mining for recommendations improvement. However, the advantages that recommender systems can take with the use of opinion mining are not exhausted yet and we found some improvement opportunities for future research.

We verified that there are methods of context-aware recommender systems that model user preferences at the item level and do not consider item aspects, i.e., they only consider contextual information related to the overall evaluation of the item [94,97,98] (Table 2). However, preferences can also be modeled at the aspect level and such preferences may be influenced by contextual factors [19,91,93,96] (Tables 1 and 3). For example, a user may put more emphasis on "location"

when his context is "business travel", but in the context of "family travel", the aspect "bedroom" may become more important. In addition, some methods that consider both contextual information and opinion information do not yet use automatic methods to extract both types of information. Therefore, we consider that the proposal of automatic methods for extracting contextual and aspect level opinion information is very relevant. Furthermore, most of the current research uses reviews as the source of contextual and opinion information (Table 4) and we believe that other types of user-generated content would be of great value to recommender systems.

We present in the next section a more detailed analysis of each paper selected in this systematic review.

4.2. Details of Selected Studies

In this section, we describe and discuss the 17 selected studies in chronological order. Our discussion considers how information is extracted from opinions and context and how this information is used and combined by the recommendation systems. Furthermore, the recommender system evaluation approaches and metrics used in each study are presented.

Ho et al. [90] proposed an approach to mining future spatiotemporal events from news articles and thus to provide information to a location-aware recommender system. According to the researchers, an event is extracted only when its location and time can be identified or inferred and, in addition to such information, the event sentiments are extracted, i.e., an event is classified as positive, negative or neutral.

According to Ho et al. [90], an event consists of six attributes: spatial (name, longitude, latitude), temporal (day, month, year, time [interval]—if available), key phrases (text before and after a temporal pattern), sentiment, information source (URL) and title of the news article. The proposed approach to mine events consists of four subtasks:

- Recognition of future and near past temporal patterns: in this task, both absolute time and relative time are treated. The considered temporal patterns are related to the future and the near past with respect to a reference time, which is, in this case, the timestamp of the article publication date;
- Toponym recognition and resolution: the main idea is the definition of a local spatial lexicon consisting of a set of toponyms of close proximity, attached to a news source. Ho et al. [90] used a hybrid technique of toponym recognition consisting of *Part-Of-Speech* (*POS*) tagging, named entity recognition (NER) and rule-based heuristic recognition, followed by matching phrases from the GeoNames gazetteer [110];
- Spatiotemporal disambiguation and matching: it is necessary to form pairs of toponyms with future temporal patterns to establish the existence of a future event. This matching process is defined by a function $f : X \longrightarrow Y$, where X is the set of future temporal patterns and Y is the set of toponyms;
- Sentiment analysis of events: task that determines the user sentiment to an identified event. Two classification approaches are applied in the bag-of-words extracted from the news articles for sentiment event classification: "*supervised Latent Dirichlet Allocation*" (*sLDA*) and "*Support Vector Machine*" (*SVM*). Positive articles have news related to topics such as festivals, entertainment and sports. Negative articles have news related to topics such as crime, accidents, bad weather and traffic. The rest is included in the neutral category. A recommender system can then advise a user to avoid a geographical location or to attend a future event based on the sentiment of the event.

Evaluation: The evaluation conducted by Ho et al. [90] consisted of comparing the accuracy values obtained with each of the two methods used to classify sentiments (sLDA and SVM). The objective of the experiments was to evaluate the performance of the sentiment classification. The authors did not conduct experiments applying the method in recommender systems, which does not allow for the analysis of results in this context.

The following points were observed in the method of Ho et al. [90]: (i) the sentiment analysis performed is at document level, i.e., if there are more than one event in the news and they have different

sentiment polarities, only the overall sentiment of the document will be considered—sentence or aspect level sentiment analysis could be more efficient; (ii) the authors did not consider the preferences of the users only their location—the preferences of the users are very relevant in recommender systems, often indispensable; and (iii) the contextual information includes only location and time—in case of recommendations of local events, it would also be interesting to know which company the user is with, which may also interfere with their choices and preferences.

Levi et al. [91] focused on recommendation of hotels. According to them, users do not rate enough hotels to enable collaborative filtering, so they design a cold start hotel recommender system that uses the text of the reviews as the main data. The contextual information is mined from review texts and analyzed for common traits per context groups. The main idea of the system is to give more importance to reviews of people with the same context. Levi et al. [91] define three types of contextual information:(i) trip intent, which can be business trip, single traveler on vacation, family, group or couple; (ii) nationality; and (ii) user preferences for the different hotel aspects. The aspects are mined from reviews using an unsupervised clustering algorithm.

In the pre-processing phase, there are three subtasks: (i) assigning weights to features for each intent; (ii) assigning weights to features per nationality; and (iii) clustering hotel features per aspect.

To find common traits for each context group, the nouns and noun phrases (features) are extracted from all reviews and those that are more common for that group are found. For each feature, a weight is assigned per context, according to their frequency in reviews within that context. The common traits of context groups are the highest weight features for that group.

To construct common traits of hotel aspects, Levi et al. [91] clustered the features upon co-occurrence in the same sentence. Each cluster contains the most relevant vocabulary for that aspect. Still in the pre-processing phase, an opinion lexicon is built to give each feature an orientation score. Once the user declares his/her context and preferences, the set of weights is chosen, corresponding to the subtasks: select relevant feature weight for intent and select relevant feature weight for nationality. An orientation score is assigned to each feature. The features, their weights and orientations are combined to build a score for each sentence. The overall score for each review is the combination of the sentence scores. The final score for each hotel is an average of all of its reviews.

Levi et al. [91]'s approach extracts key features for each group. The weight of a feature is calculated based on its frequency in sentences appearing in reviews that belong to a specific context.

The aspects are selected using clustering analysis performed on the text. The authors use an unsupervised community detection technique based on [106]. A network graph is constructed, in which each node corresponds to a feature and each community correspond to a hotel aspect. *PMI-Pointwise mutual information* weight is also used. It measures the information overlapping between two random variables. Then, weights are assigned to aspect related features.

In the feature opinion orientation step, all the adjectives that appear in the same sentence for each feature are extracted. Levi et al. [91] use a bootstrapping lexicon-based approach proposed by Hu and Liu [70]. They create, manually, a set of seed adjectives from the opinion lexicon list with semantic orientation. Then, for each adjective in the seed list, they search for a synonym and an antonym in *WordNet* [111]. Each adjective found in the opinion lexicon is assigned an orientation, and is added to the seed list. Opinion rules are also used, such as negation rule (words or phrases like "no", "not", and so on) and but clause rule (sentences containing "but", "with the exception of", "except for" and so on). Then, an orientation score is assigned to each feature in a specific sentence. When many opinion words surround a single feature, they are aggregated.

The final weight assigned to the feature for the user in their context is the multiplication of the intent weight, the nationality weight and the weight based on aspect preferences. To produce a score for each sentence, Levi et al. [91] multiply each feature by its orientation score and sum up the weight scores of all features in a sentence. To produce a score for a review, they sum up the scores of all the sentences in a review. Finally, they produced a score for each hotel, making an average of the scores of all of its reviews.

Evaluation: Levi et al. [91] conducted two types of evaluations. In the first evaluation, the authors used two datasets—one extracted from the website *TripAdvisor* and one extracted from the website *Venere*. The data contains general information about each hotel such as "name", "address", "average rating", "star" and "price", and a list of reviews written by hotel customers. The reviews include the author's travel purpose, nationality, rating, review text and "additional metadata". Levi et al. [91] designed an online experiment with *Amazon Mechanical Turk workers* [112] in which the texts of reviews were made available for each worker. These workers estimated the rating that the review's author gave to the hotel. The difference between the estimated rating and the actual rating was calculated. In the second evaluation, Levi et al. [91] used the methodology described in [113]. Such methodology does not measure the absolute satisfaction of the user, but his/her satisfaction relative to one system comparing with another. Levi et al. [91] made the proposed system online and asked friends and colleagues to evaluate it. Users entered some search parameters such as travel purpose, nationality, preference aspect, and price range. Then, a list of six hotels was presented to each user. Some of these hotels have been recommended by the Levi et al.'s system and the others were the best evaluated hotels in *Venere* and *TripAdvisor* websites. The users did not have the information of the origin of each recommendation, that is, which system had generated it, so that this did not influence the evaluation. For each of the recommended hotels, the evaluators answered whether or not they would select the hotel and gave a rating in the scale [1–5] to indicate if the recommendation met the search criteria. In addition, they also indicated which aspect most influenced the decision.

A point observed in the work of Levi et al. [91] refers to the way of obtaining contextual information. Such information needs to be informed by the user. As you know, users are not always willing to fill out forms, answer questions or inform their data. In this way, it would be more interesting for this system to use some automatic method for extracting contextual information. In addition, the way in which the method was evaluated was very subjective. An empirical evaluation would be interesting, following some evaluation protocol to obtain more objective data.

Meehan et al. [92] investigated the hypothesis that all relevant information available, which includes contextual information, should be considered in the recommendation. The authors study the proposal of a method for context-aware recommender systems for the tourism domain. The contextual information considered in the work is:

- Location: *GPS*, *GSM* and *Wi-Fi* are technologies used to obtain location information.
- Weather: the weather data are collected from *WorldWeatherOnline API* [114].
- Time: it is important to know if a point of interest is open before recommending it. Furthermore, the amount of time that a user stays in each attraction can be used to determine his/her attraction interest level.
- Social media sentiment: the sentiment analysis is executed over real-time tweets by AlchemyAPI to determine the current sentiment about the touristic point.
- Personalization: some data from social networks as age, gender, relationship status and the number of children.

The hybrid system model of Meehan et al. [92] uses several intelligent techniques to generate the recommendations: (i) neural networks, to determine the weight of each type of context for each user; (ii) fuzzy logic, to represent the interest of the user in each item in the current climatic conditions; and (iii) principal component analysis, is used to reduce the dimensionality of any traveled distance.

Meehan et al. [92] plans to use the proposed method in the mobile application VISIT (*Virtual Intelligent System for Informing Tourist*). The way in which the method will be deployed in the recommender system is not detailed nor is any empirical evaluation presented. Analyzing the steps of the method, it is noticed that the personalization may not be successful due to the fact of the users do not always make their data available in the social networks.

Chen and Chen [93] considered two kinds of user preferences: context-dependent and context-independent. The context-dependent preferences refer to the aspect level contextual needs that are common to users who are under the same context. The context-independent preferences are

relatively less sensitive to contextual changes and reflect more stable requirements for item aspects over time. Chen and Chen [93] implemented an automatic method for mining contextual opinion tuples from reviews. A contextual opinion tuple is formally denoted as $\{\langle i, rev_{u,i}, a_k, Con_{i,k} \rangle | 1 \leq k \leq K\}$, where i is the item, $rev_{u,i}$ is the review written by user u about the item i, a_k is the opinion of the user u about the aspect k and $Con_{i,k}$ is the context vector whose element value equals 1 when the associated context occurs and 0 otherwise. After extracting the context tuples, the two types of preferences are detected. Then, the context-independent and context-dependent preferences are combined via a multiplicative approach for generating a recommendation to the target user.

Chen and Chen [93] propose a synthetic method to perform contextual review analysis for extracting contextual opinion tuples. This method consists of four steps:

- Aspect identification: in this task, the relevant terms for each aspect are identified. The authors adopt the bootstrapping method proposed in [107]. In this method, each aspect is first equipped with a set of manually-selected keywords, and the other related terms are searched out through measuring the dependency between the aspect and the candidate terms based on *Chi-square statistic* [115]. Chen and Chen [93] define five major aspects: "Value", "Food", "Atmosphere", "Service", and "Location", since the reviews are about restaurants. Only frequent nouns and noun phrases are considered as term candidates. These terms are extracted by using a *Part-of-Speech* (*POS*) tagger.
- Opinion detection: in this task, the POS tagger is used to extract the adjectives in the review. Their sentiment polarity is determined with an opinion lexicon [116]. Using a distance-based score, the authors summarize all opinions expressed in one sentence.
- Context extraction: to extract contexts, a keyword matching method is employed. The authors consider that the contextual variables are "Time", "Occasion", and "Companion". Each contextual variable can be assigned with different values, and each value can be defined by a set of manually-selected keywords. If any of the keywords appear in a review sentence, the sentence will be tagged with the corresponding contextual value.
- Aspect-context relation construction: the authors follow the rules: (i) aspect level opinion and context are related if both occur in the same sentence; and (ii) the opinion is related to contextual values that occur in the previous, nearest sentence, if the sentence only contains aspect level opinion without mentioning context. The opinion a_k in tuple $\langle i, rev_{u,i}, a_k, Con_{i,k} \rangle$ is the aggregation of opinion scores of aspect-related terms that are under the same context $Con_{i,k}$.

To detect the context-independent preferences, the linear *least-square regression function* with the *statistical t-test* is used to analyze the user's history data. Each review written by the user can be represented as a rating vector $\langle a_1, \ldots, a_K \rangle$ on the set of K aspects without considering their relations with contextual factors. All the rating vectors can be used to construct the *linear least-square regression function*.

Chen and Chen [93] proposed three variations of contextual weighting methods for assigning weights to aspects in different contexts. According to the authors, the aspects' weights can be calculated by analyzing the relation between the aspect's frequency and the context. To consider the importance of the aspect-related term in different contexts, they proposed three feature selection methods for identifying the context-dependent weights of aspect-related terms:

- *Mutual information*: is used to measure the mutual dependence between aspect-related term and context.
- *Information gain*: can be applied to measure the importance of an aspect-related term to a specific context.
- *Chi-square statistic*: can measure the lack of independence between an aspect-related term and context by computing the variance between the sample distribution and chi-square distribution [115].

Then, the weights of the aspect-related terms are incorporated into the calculation of the aspect weights considering each contextual information. The context-independent preferences and the context-dependent preferences are combined to compute a score of the review. Finally, the score of an item is calculated by averaging the scores of all of its reviews.

Evaluation: Chen and Chen [93] used two real-life restaurant datasets, one from *TripAdvisor* and another from *Yelp* [117], which was published by the *RecSys* 2013 challenge. The adopted evaluation procedure was the *per-user evaluation schema*, used in [118,119]. The used metrics used were *Hit Ratio* and *Mean Reciprocal Rank*. The *Student t-test* was applied to compute the statistical significance of the difference between the compared methods. The baselines considered were:

- *Context Freer*: method proposed in [120] that does not consider the context-dependent preferences.
- *Context Pre-filter*: only the scores derived from reviews written under the target user's contexts are considered for calculating the item's score.
- *Default Connecter*: similar to the method proposed in [91]. It makes no distinction among users' opinions for the same aspect in different contexts.
- *Discriminative Connecter*: this method is also similar to the one proposed in [91]. It does not consider the weights of aspect-related terms.

The results of the experiments showed that the Chen and Chen [93]'s method was significantly better than the baselines, the *Chi-square* method being the best one. Two points in this method can be improved, such as, (i) in the aspect identification method, it is necessary to define a set of key aspects, that is, it is a domain dependent method and requires previous information. It would be interesting to use some automatic aspect identification method; and (ii) in the context extraction method, the possible context values are defined manually, that is, they are pre-defined information related to the domain—in this case, an automatic method of context extraction could be used.

Chen and Chen [19] is an extension of Chen and Chen [93]. Chen and Chen [19] implemented a *stochastic gradient descent learning method* to automatically integrate users' contextual preferences into the recommendation process. As presented in [93], the authors determine the context-dependent preferences using *Mutual Information*, *Information Gain* and *Chi-square Statistic*. To extract context-independent preferences, Chen and Chen [19] consider different properties between new users and repeated users. For new users, those with few historical records in the system, they apply the *probabilistic regression model* that can detect the preferences of new users by treating the detection as a *Bayesian learning process*. For repeated users, those with abundant history data, they compare the effectiveness of the *probabilistic regression model* and the *linear regression model*, as the latter can be used to detect users' preferences in a rich data condition. Finally, Chen and Chen [19] propose a *linear regression based algorithm* that uses a *stochastic gradient descent learning procedure* to automatically combine the two types of user preferences into the recommendation process.

According to Chen and Chen [19], their study can be considered an extension of the contextual pre-filtering based approach, as it also first filters out ratings according to the target user's contexts and then generates recommendations, but the difference is that their pre-filtering approach is conducted at the aspect level instead of the item level. The steps are almost the same proposed in Chen and Chen [93]:

- Contextual opinion extraction (extracting contextual opinion tuples from reviews): this task consists of transforming user-generated reviews into structured contextual opinion tuples. The methodology used is the same of Chen and Chen [93]. The steps were already previously presented.
- Inferring context-independent preferences: Chen and Chen [19] consider two alternative inference models: the *linear regression model* and the *probabilistic regression model*. The *linear regression model* assumes that a user's overall evaluation of an item is the sum of his/her opinions about different aspects of the item, so it can be generated by aggregating the aspect level opinions. To use the *probabilistic regression model*, the relation between the overall rating and all aspects' opinions must

be essentially a regression problem. PRM models the underlying relation via Bayesian treatment so that prior knowledge can be incorporated into the model.

- Inferring context-dependent preferences: context-dependent preferences indicate the aspect level contextual needs that are common to users in the same context. To capture such preferences, the same method proposed by Chen and Chen [93] is used.
- Recommendation generation process: is almost the same presented in [93]. Chen and Chen [19] implement a *linear-regression-based method* to combine context-dependent preferences and context-independent preferences when computing a score for a review written by the target user. The difference is that, in this extension, they propose a *stochastic gradient descent learning method* to learn the combination parameter automatically. This parameter is used to control the relative contributions of a user's context-independent and context-dependent preferences for an aspect in a specific context value, when computing a review's score.

Evaluation: The used datasets and the adopted evaluation procedure are the same of Chen and Chen [93]. They used two real-life datasets, one from *TripAdvisor* and other from *Yelp*, but differently from Chen and Chen [93], they also considered the hotel domain. The baselines considered are three out of the four baselines used in [93]. Chen and Chen [19] compared their algorithms against three related methods: (i) context free; (ii) context pre-filter; and (iii) simple connecter. The evaluation metrics used were *Hit Ratio* and *Mean Reciprocal Rank*. The evaluation was divided into three parts: experiment with new users, experiment with recurrent users and experiment with all data. The results led the authors to three conclusions: (i) the probabilistic regression model is adequate to infer the context-independent preferences of both new users and recurrent users; (ii) to detect context-dependent preferences, the *Chi-square* method is the most appropriate; and (iii) the stochastic gradient descent learning method, in addition to learning the combination parameters, improves the accuracy of the recommendation.

Because the method of Chen and Chen [19] is an extension of the method of Chen and Chen [93], the critical analysis is the same as the previous work. There are possibilities for improvements, already discussed, in the aspect identification and context extraction methods.

Colace et al. [94,95] proposed a collaborative and user-centered approach that provides social recommendations, capturing and exploiting users' opinions and sentiments about items. In their approach, several aspects related to users are considered and integrated together with items' features and context information within a general recommendation framework. These aspects can be preferences, opinions, behavior and feedback.

In the approach of Colace et al. [94,95], a pre-filtering process is conducted, in which a subset of items that are good candidates to be recommended is selected. To select these items, the first step consists in clustering together similar items, where the similarity should consider all the different spaces of features. Colace et al. [94,95] employ *high-order star-structured co-clustering techniques* to address the problem of heterogeneous data pre-filtering.

In their recommendation problem, a user is represented as a set of vectors in the same feature spaces describing the items. The cosine distance is used to provide a first candidate list of items to be recommended. This distance is calculated between the user vectors and the centroids of each item cluster. The most similar item cluster is chosen. To provide the pre-filtered list of candidate items, Colace et al. [94,95] adopt one of two strategies: (i) set-union strategy, in which the items belonging to the union of all clusters are selected; or (ii) threshold strategy, in which the items that appear in at least a given number of clusters are selected. Finally, items already visited/liked/browsed by the user are filtered out.

According to Colace et al. [94,95], when an item is chosen after another item in the same user browsing session, this event means that the second item is voting for the first item. Colace et al. [94,95] also say that, if an item is very similar in terms of some intrinsic features to another item, this can also be interpreted as one recommending the other. Therefore, the browsing system for a set of items is modeled as a labeled and directed graph. Each edge of the graph is associated with two variables:

one variable that indicates the type of the edge (pattern or similarity) and the other variable is the weight of the edge. A pattern label for an edge denotes the fact that an item was chosen immediately after another item, so, in this case, the weight of the edge is the number of times this fact occurred. A similarity label for an edge denotes the fact that an item is similar to another item, so the weight of this edge is the similarity between the two items. From the graph, Colace et al. [94,95] calculate the recommendation grade of a given item.

The sentiment extraction technique used by Colace et al. [94,95] is an improvement of the approach presented by Colace et al. [108], where the *Latent Dirichlet Allocation* has been adopted for mining the sentiment inside documents. According to Colace et al. [94,95], the knowledge within a set of documents can be represented by using a *Mixed Graph of Terms* (*mGT*). In this way the *mGT* contains words (and their probabilistic relationships) which are representative of a certain sentiment for that knowledge domain. The algorithms work as follows: for each word in the positive *mGT* (obtained analyzing the positively opinionated items) and the negative *mGT* (obtained analyzing the negatively opinionated items), their synonyms are retrieved through the annotated lexicon, more specifically an enriched version of *Wordnet* [121,122]. Each review is analyzed and the system discards those whose trustiness is below a threshold. The negative and positive probabilities are calculated and weighted by the use of a correction factor, which takes into account reviewer's trustiness and rating. The probabilities are determined for each item, which expresses the probability that a sentiment, extracted from the set of comments related to a given item, is "positive" or "negative". Finally, Colace et al. [94,95] define a ranking refining function that combines the recommendation grade of an item and the values of the negative and positive probabilities. The output of this function is, for each item, the final ranking value. This function increases the recommendation grade value if the sentiment within item's comments is positive, and decreases it in the case of negative sentiment.

Finally, in the post-filtering stage, the contextual information is used to generate the final set of candidates for recommendation. Colace et al. [94,95] represent the context by means of the *key-value model* [52] using as dimensions some of the different feature spaces related to items. If a user is accessing an item, the set of recommendation candidates includes the items that have been accessed by at least one user within a given number of steps from the actual item and the items that are most similar to this item according to the results of a *Nearest Neighbor Query* functionality. The ranked list of recommendations is then generated by ranking the candidate items, obtaining the final set. All the items that do not respect possible context constraints for each user are removed from the final list.

Evaluation: the evaluation was performed on two system variables proposed by Colace et al. [94,95]—(i) using the system to recommend travel packages, in which the considered dataset was collected from site *TripAdvisor*. It consists of a subset of approximately 5000 travel-related items. Colace et al. [94,95] asked a group of about 50 people to explore a collection of travel items and complete 20 tasks using *TripAdvisor*. The authors used two strategies to evaluate the results, empirical measures of access complexity that consider mouse clicks and time, and TLX (*NASA Task Load Index factor*); (ii) using the system to recommend movies, in which more than 10,000 items from the IMDB website were collected. Colace et al. [94,95] adopted an evaluation strategy that aims to measure user satisfaction with search package travel tasks and the effectiveness of the system in terms of accuracy for the movie recommendation problem. The metrics used were *Mean Absolute Error* and *Root Mean Square Error*. The baselines of the experiments were a user-based collaborative approach and an item based collaborative approach, both described in [123].

Some difficulties can be identified in the use of the method proposed by Colace et al. [94,95]: (i) it is necessary to have the information of accesses or purchases of the users, since the information of which item was accessed/purchased before or after another item is needed; and (ii) the trustiness information of the user is required.

Kothari and Patel [96] considered that context is any constraint or condition passed by the user. According to them, there are two types of recommender systems: context independent and context dependent.

The authors state that systems that consider only general evaluations of reviews (for example, the stars) are not adequate to generate accurate recommendations for users. For Kothari and Patel [96], it is also necessary to consider the influence that context exerts on user aspect level preferences.

Kothari and Patel [96] proposed a recommender system that consists of an automatic detection of context-aspect relations of reviews. Moreover, this system combines the context-independent and context-dependent preferences of users. The *Support Vector Machine* (SVM) technique is incorporated into the system to classify user preferences in their respective contexts. The steps of the methodology proposed by Kothari and Patel [96] are:

1. The data is preprocessed and cleaned by stemming terms and removing noise data and irrelevant reviews.
2. The opinion tuples are extracted:

    - the aspects are identified;
    - the opinion value or sentiment polarity are identified;
    - the context parameters are found and their possible values are defined;
    - finally, the opinion tuples formed by aspects and contexts are constructed.

3. The context-independent preferences are filtered using least-squares linear regression.
4. The context-dependent attributes are filtered using the methods *Gain Information* and *Chi-square Statistics*.
5. The attributes resulting from the previous steps are applied as input vectors in the SVM model.
6. The classified data are used for the recommendation process that consists of a collaborative filtering technique.

Evaluation: in the experiments, Kothari and Patel [96] used real datasets extracted from the *TripAdvisor* site. The metrics applied were *Hit Ratio* and *Mean Reciprocal Rank*. Individual use of the *Information Gain*, *Chi-square Statistics*, and *Mutual Information* techniques were compared to the joint use of these techniques by using SVM. The results demonstrated that these feature selection methods combined with collaborative filtering was a good strategy. Moreover, the use of these methods along with SVM and the use of SVM along with collaborative filtering have yielded good results.

Orellana et al. [97] proposed an approach that automatically extracts affective context from user comments associated with short films. The authors explored two approaches for the association of emotions with films of this category:

- They obtained annotations made by a group of people from *Amazon Mechanical Turk*.
- Orellana et al. [97] automatically extracted affective context from the comments available on *Youtube* [124] and evaluated its importance by applying this context in an emotion-aware recommendation task.

Orellana et al. [97] used a collection of short films that participated in two festivals in which *Youtube* was used as a dissemination platform: *Tropfest* and *Your Film Festival*. The *Youtube's Data API* [125] tool was used to collect a set of participating short films. For each movie the number of views, likes, dislikes and the user comments on the movie were collected. The dataset consists of 235 short films and a total of 21043 comments. The users who participated in the annotations answered some polarity questions that indicate the emotions of each film. The emotions considered by Orellana et al. [97] were: joy–sadness, anger–fear, trust–disgust, and anticipation–surprise. The authors associated each answer with a numerical value. Users also indicated whether or not they liked the films and labeled each movie in the contexts:

- Audience: children, adolescents, young people, adults, the elderly and the whole public.
- Companion: friends, family, partner, alone and anybody.
- Time: to relax after work, during a break at work, for entertainment during weekends or on vacation, and at anytime.

Orellana et al. [97] explored how emotions are distributed for both the liked and disliked films. In addition to this exploration, the authors proposed the *Automatic Emotion eXtraction* (AEX) approach that explores the comments of the *Youtube*, associated with short films, to detect the emotions they evoke. The steps of the approach are:

1. A profile is built for each movie, which is made up of all user comments about the movie.
2. Part-of-speech annotation is performed on each profile using *LingPipe* [126] *MorphAdorner* [127], in order to extract nouns and adjectives.
3. Using the *NRC Emotion Lexicon* (EmoLex) and a term matching technique, each term is associated with values of emotions and polarities.
4. Finally, the vector of emotion and polarity is constructed.

Orellana et al. [97] incorporated the affective context into the recommender system following a collaborative ranking approach. The *LambdaMART* method was used as the learning-to-rank method.

Evaluation: The data were divided into training, validation and testing. The first 20% of users were selected for the test group. Therefore, 10% of users formed the validation set. In addition, finally, the rest of the user data is used for training. The metrics used were *Precision* and *Mean Reciprocal Rank*. Orellana et al. [97] compared the results of the recommendation using the affective context extracted by the proposed AEX method to the results of the recommendation using the affective context annotated by the users. The results showed that the system based on the AEX method is very competitive.

The method proposed by Orellana et al. [97] considers the detected emotion as a context, not taking into account other types of contextual information. In addition, the extracted emotions are predetermined and suitable for the movie domain.

Yang et al. [98] focused on contextual suggestions based on location and proposed using user's opinions to build the user's profiles. The user's profile is a representation of the user that considers all the available information about his/her. They also developed a new summary generation method that uses opinion-based user profiles to generate summaries of suggestions. According to Yang et al. [98], the process of generating contextual recommendations based on location is performed in two steps: (i) identify places of interest that are close to the target user's current location; and (ii) classify the candidate places considering the interest of the user. The authors focused on the second step. In their work [98], U represents a user, CS represents a candidate suggestion, $S(U, CS)$ represents the score of relevance between the user and the suggestion that must be estimated. For each user U, its preferences/ratings are obtained. ES is an example of suggestion and $R(U, ES)$ is the rating given by the user u to example ES.

Yang et al. [98] model the user profile by using opinions ⟨*rating*, text of *review*⟩ on the sample suggestions. They use positive reviews of the examples that the user liked to construct his/her positive profile, and use the negative reviews of the examples that the user did not like to build his/her negative profile. When a user's lack of opinions occurs, the opinions of users similar to him/her are used, that is, users who have evaluated a suggestion in a similar way.

To generate the representations of the suggestions, these being representations based on opinions, Yang et al. [98] construct two profiles: (i) positive profile based on all positive reviews about the suggestion; and (ii) negative profile based on all negative reviews about the suggestion. The researchers explored four strategies to construct the profiles of the suggestions:

- Complete reviews: this approach considers all the terms of the reviews texts about an item to construct the profile of that item.
- Selected terms from reviews: this approach constructs the profile of an item based on a set of selected terms. Yang et al. [98] considered the 100 most frequent terms in the texts of the reviews about the item.
- Nouns of reviews: this approach uses only nouns of the review texts.
- Summary of reviews: this approach uses the *Opinosis algorithm* [128] to generate concise summaries of reviews to construct the profiles.

Yang et al. [98] investigated two possible ways to combine the similarities between user profiles and representations of candidate items:

- Linear interpolation: this method combines multiple scores in a single score. Yang et al. [98] consider that relevance score can be positively correlated with similarity between two positive profiles and two negative profiles and can be negatively correlated between positive and negative profiles. To calculate the score, parameters are used to balance the impact of the components. In other words, the score of a candidate item for a user is calculated by summing the similarities between positive profiles and negative profiles and subtracting the similarities between negative and positive profiles. The similarities are multiplied by the parameters.
- Learning-to-rank: this method considers the similarities as attributes and uses Learning-to-rank methods to calculate the score of the ranking. Three methods were used: (i) *MART*, also known as *Gradient Boosted Regression Trees*; (ii) *LambdaMART*; and (iii) *LinearRegression*.

To generate a customized and structured summary for a candidate suggestion, Yang et al. [98] considered four components:

1. Introductory sentence: name of the suggestion followed by its category.
2. "Official" introduction: Yang et al. [98] first extract frequent nouns from reviews about the suggestion. These nouns are used to extract sentences from the suggested website. These sentences are classified according to the number of positive adjectives present in them and only the five best classified sentences are used to not extend the size of the summary.
3. Highlighted reviews: sentences with more positive distinct adjectives are chosen.
4. Final sentence: "We recommend this suggestion to you because you like *abc* and *xyz* in the suggestions".

Evaluation: Yang et al. [98] evaluated the proposed method using two datasets: (i) the dataset used in *TREC Contextual Suggestion* track (CS2012, CS2013 and CS2014); and (ii) a dataset extracted from the site *Yelp*. For each user, the suggestions were divided into two sets of the same size, training and test. The reviews were mapped to positive or negative according to the rating related to it. For example, in base CS2012, the rating equal to 1 was mapped to positive and the rating equal to −1 was mapped to negative. *Precision* and *Expected Reciprocal Rank* were used as evaluation metrics. The proposed method was compared to two baselines which are two methods that use different types of information to construct the user profile [129]. Such baselines construct two representations for users (positive and negative), but only one representation for the suggestions. The evaluation using linear interpolation was performed with *5-fold cross validation*. To evaluate the *Learning-to-rank* methods, the models were trained on 60% of data, validated at 20% and tested on the remaining 20%. The results showed that Yang et al. [98]'s method significantly outperforms baselines and the best strategy is to use the noun representation of the reviews and the *MART* method to match the similarities.

Yang et al. [98] consider location as the only contextual information. In addition, the polarity of each review's sentiment is defined only by the user's rating, and no sentiment analysis is performed in the text of the review.

Zhao et al. [99] proposed a model to conduct service quality evaluation using the concept of user rating's confidence. They first utilized entropy to calculate user rating's confidence. Therefore, they explore spatial-temporal features and review sentimental features of user ratings to restrict their confidence. Lastly, the authors fuse them into a unified model to calculate an overall confidence by fusing that features.

According to Zhao et al. [99], it is very important to consider quality of service in recommender systems because high-quality services should be recommended more easily. The basic idea of the work of Zhao et al. [99] is that user rating's confidence is different in different places, different times, and different sentiments.

To calculate the user rating's confidence, Zhao et al. [99] consider that, when users' ratings are confident, their ratings must have little differences with the overall rating of items. Therefore, the

information entropy of value of these differences can be used to represent the confidence value of user ratings. The lower entropy value is, the more stable the system is, and the more confident the user's rating is.

User's preferences are changing constantly, so their rating's confidence may be different according to places and times. Furthermore, sometimes, users give high ratings; however, there are many negative words in their reviews. In this way, the rating's confidence is calculated considering spatial-temporal features and reviews sentimental features.

Zhao et al. [99] analyzed the distribution of rating's confidence in different user–item geographic location distances. They found that the rating's confidence is low if users are very close to the rated items. The authors also analyzed the distribution of rating's confidence in different periods of time, and they found that it decreases over time.

With regard to sentimental features, Zhao et al. [99] used the method of sentiment analysis proposed in Zhang et al. [109]. The authors analyzed the distribution of the average difference and the corresponding number of ratings in different sentiment scores. They found that user confidence increases with review sentiment score.

The service quality evaluation model is a probabilistic model that fuses user's confidence with contextual features, including spatial-temporal features and review sentiment features, to calculate an overall confidence value of a rating.

Evaluation: Zhao et al. [99] used two datasets to evaluate the proposed approach—*Yelp* and *Douban* datasets—being the *Yelp* dataset split in *Yelp Restaurants* and *Yelp Nightlife*. They pre-selected some items to be used for training and others to be used for testing. Every tested item has at most five ratings. The evaluation metrics used were *Root Mean Square Error*, the *Mean Absolute Error*, *Precision*, *Recall* and AUC (*Area under Curve*). The authors used nine baselines to evaluate the proposed approach, named *Service Quality Evaluation* (SQE): (i) BM (*Basic Method*); (ii) *Biases* (*Basic Biases*) [35]; (iii) BT (*Biases Based on Taxonomy*) [130]; (iv) *BaseMF* [131]; (v) *CircleCon Model* [132]; (vi) *ContextoMF* [133]; (vii) PRM [134,135]; (viii) *Item-based Collaborative Filtering* [136] and (ix) *MART-SQE* (proposed approach with *multiple-additive regression trees*—MART [137]). Zhao et al. [99] evaluated the SQE analyzing its performance and the impact of six aspects—data sparsity, review count, different curves fitting approaches, different features, less training data and the type of prediction. The authors concluded that the proposed approach can use few ratings to predict the overall rating of services. Furthermore, it has wide applicability for different domains and datasets.

Analyzing the approach, we note that Zhao et al. [99] did not mention how they get the contextual information local and time. Intuitively, observing the types of contextual information, we imagine that the information was obtained from access logs or the users themselves passed this data at the time they wrote the reviews. In this way, an automatic contextual information extraction approach was not used. Furthermore, the sentiment analysis performed by the Zhao et al. [99]'s approach is not at the aspect level, which makes it impossible to understand the preferences of the users in the aspect level.

Kharrat et al. [100] proposed using external resource, i.e., Facebook comments, to mitigate the cold start problem and to improve the recommendation. The proposed approach consists of three parts:

1. In the first part, the comments posted by the users are stored in a comments' dataset. The data used was the MovieLens dataset. For gathering data from a social network, i.e., from Facebook, the authors created a script in Java. The data collected was a collection of 1000 Facebook profiles, which included comments related to the topic of movies. Using a matching approach, the authors included all MovieLens' users to the profile collection. This matching is based on demographic information about the users, like age, gender, occupation and country. Therefore, all users' profiles were stored in the dataset.
2. The second part relies on three fundamental aspects:

   a   Tags graph—the system uses a tags graph to represent item descriptions.

    b   Linguistic resources—the *WordNet* is used to construct a lexical resource for opinion mining. This resource is represented in tags form.

    c   Extraction algorithm of tags—used to annotate all tags in comments, the item tags and the opinion tags. Therefore, the contextual information consists of users' opinion for different tags of items. The users' profiles are described by opinions of several tags and are stored in the dataset.

3.   The third part is the recommendation algorithm. The systems accepts as input a comment related to a specific item and provides opinion scores for every item's tags of this item. The phases of the recommendations are:

    a   Creation of opinion's score—for every tag's opinion, a relevant score is attributed to this tag.

    b   Integration of opinion (contextual) dimension into recommendation algorithms—the new algorithms are improved versions of Slope One algorithm and Simon Funk's SVD algorithm, named SemSlope One and SemSVD, respectively.

Evaluation: Kharrat et al. [100] used MovieLens dataset with 1000 ratings by 943 users on 1682 items. Each user rated at least 20 movies. The aim of the evaluation was to compare SemSlope against Slope One and SemSVD against SVD. The used metrics were *Root Mean Square Error*, *normalized Discounted Cumulative Gain*, *Precision* and *Recall*. The dataset was partitioned in a *5-fold cross* validation configuration. For each user in each partition, 20% of their ratings are selected as test set and 80% as training set.

Although not clear, the opinion mining approach employed by the authors resembles the aspect-based opinion mining. With the difference that the aspects are the tags that describe the items. The extraction of the tags depends on a pre-defined list of item tags and the analysis of feelings for each tag is done by using a dictionary of opinion words built based on *WordNet*. Automating the extraction of the tags, which would discard the need of manually defining a set of tags, would improve the method of Kharrat et al. [100]. In addition, user opinion is considered as its context, not taking into account other types of contextual information.

Missaoui et al. [101] aimed to predict the users' preferences in the tourism domain, to provide personalized and context-aware recommendations. They proposed a Content-Based Filtering approach that identifies the nearest tourism-related services to the user and recommends the most relevant ones considering the opinions that the user has expressed through her/his previous reviews. The proposed Content-Based Filtering approach is based on four tasks:

1.   The definition of the user profiles—the authors follow the idea of positive and negative profiles proposed by Yang et al. [98], previously described. Two language models based on unigrams are defined, a positive language model and a negative language model. The positive and negative language models represent the user's positive and negative feedback, respectively. If the rating related to the review is >3, the review is positive. On the other hand, if it is ≤3, the review is negative.

2.   The definition of the Tourism-Related service profiles—in addition to the user profiles, for each service, a T-R service profile represented by a positive language model and a negative language model is built. The approach considers the reviews written by the elite *Yelp*.

3.   The comparison between the profiles of the services and the profile of the target user—first, the nearest Tourism-Related services (considering the user's geolocation) are selected as the set of potential candidate restaurants to be recommended. Therefore, to calculate the recommendation score for each service, the user profile is compared with the T-R service profiles. The recommendation problem consists of the task of calculating the similarity between the positive and negative components of the user profile and the same two components of the Tourism-Related service profile.

4.   The recommendation of the Tourism-Related services—after calculating the recommendation scores of the services closer to the target user, the top-k T-R closest services in descending order of similarity are recommended.

Evaluation: Missaoui et al. [101] evaluated the recommendation approach by using the metrics *Precision* and *normalized Discounted Cumulative Gain*. The authors carried out the experiments on the restaurant *Yelp Challenge Dataset*. They selected the T-R services belonging to the restaurant category and that received trusted reviews. Missaoui et al. [101] followed the *TestItems* methodology described bay Bellogin et al. [138]. They compared the proposed approach against two baselines—labeled UCF and labeled cat-F.

Analyzing the work of Missaoui et al. [101], we found that only a type of contextual information is used, the location of the user, which is obtained automatically, but other types of contexts could be considered. In addition, the opinion analysis is done at the document level and without any robust opinion mining technique, considering only the value of the rating given by the user. Performing an analysis in the user text can give more accurate and detailed indications about the user's opinion.

Jalan and Gawande [102] proposed a hotel context-aware recommender system that mines contextual information and sentiments from reviews. The authors state that hotel recommender systems suffers from the cold start problem because users usually do not provide adequate ratings. To solve this issue, the authors considered three types of context: (i) travel type, which can be "business", "couple", "individual", "group" and "family"; (ii) user's location; and (iii) user opinion about different hotel aspects.

The main idea of the Jalan and Gawande [102]'s system is giving more importance to user reviews belonging to the same context. The recommendation process begins with the dataset collection and preprocessing. In preprocessing, the following tasks are performed: (i) Tokenization, where the sentences are split into tokens; (ii) Stopword removal; and (iii) Word replacement, where the text is normalized using a dictionary constructed by the authors. The aim here is solving some problems of short and user texts; (iv) Stemming; (v) Part-Of-Speech tagging (POS) using OpenNLP [139]; (vi) Feature (aspect) extraction, where nouns are identified as aspects; (vii) Aspect clustering, in this step of pre-processing, a semi-supervised clustering algorithm is used to find the vocabulary of each aspect. After the pre-processing, the polarity identification is carried out, where sentiment polarities are assigned to the aspects. In this step, the authors used a lexicon-based approach, considering the adjectives presented surround the aspect. After the polarity of the aspects is identified, a weight is assigned to each aspect in each context. This weight is calculated according to the number of occurrences of the aspect terms in the reviews belonging to a given context. Finally, the final hotel score is calculated. For this calculation, a score is defined for each sentence, then for each review and then for each hotel. Hotels with higher score values are recommended to the user.

Evaluation: the experiments were conducted using a *TripAdvisor* review dataset of five hotels. Jalan and Gawande [102] did not describe the evaluation in detail; they only demonstrated through graphics the number of aspects and their sentiments extracted from reviews.

Although the work of Jalan and Gawande [102] considers aspect-based opinion mining, some points should be discussed. First, it is a work close to the work of Levi et al. [91], previously described, with some similarities with respect to the contexts considered, in addition to the way of extracting opinion and calculating scores. Thus, the "travel type" context is not extracted automatically but is informed by the user. The way the "location" context is extracted or used is not discussed in the paper. Finally, the evaluation presented by the authors did not address the performance of the recommender system.

Baral et al. [103] proposed a model called ReEL (*Review aware Explanation of Location Recommendation*) for the recommendation of Points of Interest (POIs). The ReEL exploits a deep neural network to model the review aspect correlation. Furthermore, it formulates user-aspect bipartite relation as a bipartite graph, and models the explainable recommendation by using dense subgraph extraction and ranking-based techniques.

The focus of the paper is on the recommendation explanations. Baral et al. used *Convolutional Neural Networks* (CNN) to formulate the correlation problem of reviews and aspects. They map users'

sentiments to tuples (POI, aspect) and model users' aspect preferences like the aspect-POI bipartite relation. The bipartite relationship is represented using a bipartite graph.

First, the reviews are classified. The classification module executes the pre-processing, where the reviews are divided into sentences and the stopwords are removed. The second step is the aspect extraction, in which nouns and noun phrases are filtered using a frequency-based approach and then a rule-based approach is used to capture other aspects terms. After extracting the aspect terms, the terms are categorized into aspects. The *WordNet* synsets are used in this step. Next, the preparation of the sentence-aspect training data is done. The text of the review is labeled according to the aspect closest to the terms of aspects of the review. An aspect term can have three associated aspects. Sentences can have multiple labels. These labeled data are used in CNN-based sentence-aspect classifier training. The classification module classifies a review sentence into a relevant aspect.

The CNN's entry is a word embedding of the review sentences built with *Word2Vec*'s assistance. The output of the classifier is a bipartite relationship between reviews and aspects. For each user, the classifier generates a set of user feature vectors (embedding on his/her preferred aspects). The same is done for POIs. The sentiments of each review sentence are extracted using the trigram around the aspect terms. The embedding of the sentiment term is concatenated to the POI feature vector.

The proposed recommender is called *Deep Aspect-based POI recommender* (DAP). Baral et al. [103] add other types of contexts to the POI feature vector, such as categorical, spatial and others. The recommendation problem is then formulated as a matrix whose rows represent user, POI, and elements from different contexts. In this way, the recommendations are generated by a factorization machine. The authors proposed three different methods to generate the explanations of the recommendations, using a bipartite graph that represents the relation POI-aspect.

Evaluation: Baral et al. [103] used three datasets (*Yelp*, *TripAdvisor* and *AirBnB* [140]. Four models were evaluated: one model that does not present explanations for the recommendations and three models representing each approach proposed by them to generate explanations. In addition, the modules of aspect extraction, aspect categorization and sentence-aspect classification were also evaluated. The authors considered eight baselines. In the experiments, they used *5-fold cross validation* and as evaluation metrics they used *Precision*, *Recall* and *F-score*. The results showed that the proposed methods outperformed the ones without explanation, and gained significant improvement.

The work of Baral et al. [103] is very interesting, whose focus is on the explanations of the recommendations. The authors use contextual information as well as opinion mining at the aspect level. However, they do not tell how the other types of contextual information such as categorical and spatial are extracted/acquired. However, they demonstrated that the proposed method of explaining the recommendations yielded better results for the recommendation.

Sulthana and Ramasamy [104] proposed an *Ontology and Context Based Recommendation System* (OCBRS) for the book domain that uses the *Neuro-Fuzzy Classification* approach, also proposed by them. This approach develops a set of fuzzy rules to classify the reviews and to extract contextual information.

The steps of the OCBRS are: (i) the book reviews from *Amazon* are collected and stored; and, (ii) according to Sulthana and Ramasamy [104], context is a term in the review that indicates the qualitative attributes of the product or circumstance of the reviewer. They consider that the context is a noun that is accompanied by an adjective. In this way, the *Stanford Parser* [141] is used to identify nouns, adjectives and adverbs. The system uses *SentiWordNet* [142] to identify the polarity of the review sentences. To extract the context and its relation with the opinion word, the authors use *Neuro-Fuzzy Classification*. The context extracted from the review refers to the noun or verb and its relation to the word of opinion; (iii) the context review ontology is constructed to store the context of products in the domain; and (iv) the context from the ontology is given as input to the recommender system.

Evaluation: Sulthana and Ramasamy [104] used an *Amazon* book review dataset. This dataset was collected from the *Amazon* site between December 2010 and March 2015. As baselines used two

systems, the system proposed in [143] and the system proposed in [135]. According to the results, the OCBRS performed better than the baselines.

The context description given by Sulthana and Ramasamy [104] is closely related to the aspect definition and the sentiments related to them. Thus, the authors consider in their work only the opinion as contextual information.

Zangerle et al. [105] proposed methods for music recommendation. To do this, they used tweets called #nowplaying, which are tweets in which users describe the music they are listening to. Generally, in these tweets, users use hashtags to describe their emotional states. The authors consider the user's emotional state as contextual information. In this way, only tweets containing hashtags representing emotions were used.

To detect sentiments from tweets, Zangerle et al. [105] used a method known as sentiment lexica. Therefore, four opinion dictionaries were used: *AFINN* [144], *Opinion Lexicon* [70], *SentiStrengh* [145] and Vader [146]. First, the hashtag and dictionary are matched. For hashtags that are not matched, the lemmatization is applied, and then the hashtags and dictionary are matched. For cases of compound hashtags, they are split by considering the upper-case characters as boundaries between the terms and then matching them with the dictionary.

After computing affection, latent attributes are computed. For this, Zangerle et al. [105] construct a graph containing three types of objects, users, tracks and hashtags. Therefore, they use a network embedding algorithm, DeepWalk, to learn the representations of these objects.

To rank the tracks, Zangerle et al. [105] used latent attribute representations built by the network embedding technique. The authors proposed seven methods to rank the tracks.

Evaluation: Zangerle et al. [105] conducted three experiments: (i) evaluation of the effectiveness of latent attributes; (ii) evaluation of the effectiveness of affection and hashtag information; and (iii) evaluation of the effectiveness of individual sentiment lexica, as the dataset used was the #nowplaying tweets dataset compiled by Zangerle et al. [105]. The same dataset was preprocessed and transformed into two. In the first stage of the preprocessing, the tweets were removed without feeling information, which resulted in a base of 560,000. In the second step, a method of removal of outliers was applied, resulting in a base of 90,000. *Mean Reciprocal Rank* was used as the evaluation metric. The results showed that the affective information is able to improve the recommendation.

Zangerle et al. [105] consider the user's emotional state as the only contextual information. The information is extracted using a technique of sentiment analysis, but it is not an aspect-based technique. Thus, the hashtag that demonstrates the user's emotional state is extracted and, when it is a sentiment word, its polarity is defined by means of an opinion lexicon.

In Table 5, we present the strengths and weaknesses that we observed in the works discussed in this section.

Table 5. Strengths and weaknesses overview of the studies analyzed in the systematic review.

Reference	Strengths	Weaknesses
Ho et al. [90]	• contextual information is extracted automatically.	• document level opinion mining; • it does not consider the user preferences; • only location and time as contextual information.
Levi et al. [91]	• it uses unsupervised clustering to build a vocabulary for hotel aspects; • aspect level opinion mining.	• contextual information needs to be informed by the user.
Meehan et al. [92]	• contextual information is extracted automatically; • it considers many types of contextual information.	• the personalization step may not be successful due to the fact that the users do not always make their data available in the social networks.
Chen and Chen [19,93]	• aspect level opinion mining; • contextual information is extracted automatically.	• it is necessary to define a set of key aspects; • the possible context values are defined manually.
Colace et al. [94,95]	• it considers many aspects related to users together with item's features and contextual information; • it can be applied in different domains.	• it is necessary to have the information of accesses or purchases of the users; • the trustworthy information of the user is required; • opinion mining is not applied at the aspect level.
Kothari and Patel [96]	• aspect level opinion mining; • contextual information is extracted automatically.	• it is necessary to define a set of key aspects; • the possible context values are defined manually.
Orellana et al. [97]	• contextual information is extracted automatically.	• only emotion as contextual information; • the emotions have to be predetermined; • opinion mining is not applied at the aspect level.
Yang et al. [98]	• opinion-based user profile is constructed in a collaborative way; • it generates personalized summaries for the suggestions.	• only location as contextual information; • the sentiment is defined only by the user's rating.
Zhao et al. [99]	• it can use a few ratings to predict the overall rating of services.	• document level opinion mining; • contextual information is not extracted automatically.
Kharrat et al. [100]	• contextual information is extracted automatically; • aspect-based opinion mining.	• the aspects are extracted using a predefined list of item tags; • only opinion as contextual information.
Missaoui et al. [101]	• contextual information is extracted automatically; • it can be applied to a variety of similar recommendation tasks.	• only location as contextual information; • the sentiment is defined only by the user's rating.
Jalan and Gawande [102]	• aspect-based opinion mining.	• contextual information needs to be informed by the user.
Baral et al. [103]	• aspect-based opinion mining; • it uses multiple contexts; • it generates explanations.	• it does not explain how the contextual information was extracted.
Sulthana and Ramasamy [104]	• contextual information is extracted automatically; • aspect-based opinion mining.	• only opinion as contextual information.
Zangerle et al. [105]	• contextual information is extracted automatically; • it applies an unsupervised sentiment dictionary approach.	• it does not apply aspect level sentiment analysis; • only emotion state as contextual information.

5. Conclusions

Recommender systems aim to assist users to choose the item that best meets their needs in a set of many choices. Context-aware recommender systems also consider, besides the user's preference history, contextual information to make recommendations. Studies have been proving that context-aware recommender systems present better results than traditional recommender systems, which do not consider extra information to make recommendations. With the Web 2.0, more and more content have been being created by the users, like the reviews, which offers rich information to be used by recommender systems. The information obtained from user-generated content can benefit recommender systems because, for instance, it can help to deal with the problem of large data sparsity and to solve the cold-start problem for new users. A user opinion about an item (product, service or subject) reveals his/her dissatisfaction or satisfaction on it and how much he/she cares about certain characteristics of the item. For this reason, opinion mining provides valuable information for recommendation systems, especially if combined with contextual information.

Context-aware recommender systems and opinion mining are recent research topics, with an increasing number of publications. Motivated by the advances in these two research fields and the potential contributions of opinion information to recommender systems, we conducted a systematic review on context-aware recommender systems that use opinion mining, i.e., systems that consider the user's opinion about an item and/or about item aspects in addition to contextual information. The main contributions of our work are: (i) it identifies the research on context-aware recommender systems that also use information extracted by opinion mining; (ii) it maps how this research is combining these two technologies (context-aware recommender systems and opinion mining); (iii) it points out some areas that may be improved in future primary works, as well as open research challenges; (iv) its conduction followed a well-defined literature review protocol; and (v) its results may be of great help for researchers working with both opinion mining and context-aware recommender systems. Thus, this work filled a gap in the literature as, to the best of our knowledge, this is the first literature review of this broad subject.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results'.

References

1. Ricci, F.; Rokach, L.; Shapira, B. *Recommender Systems Handbook*, 2nd ed.; Springer Publishing Company: Berlin/Heidelberg, Germany, 2015.
2. Amazon. Available online: https://www.amazon.com (accessed on 21 January 2019).
3. Linden, G.; Smith, B.; York, J. Amazon.com Recommendations: Item-to-Item Collaborative Filtering. *IEEE Internet Comput.* **2003**, *7*, 76–80. [CrossRef]
4. Netflix. Available online: https://www.netflix.com (accessed on 21 January 2019).
5. Last.fm. Available online: http://www.last.fm (accessed on 21 January 2019).
6. TripAdvisor. Available online: https://www.tripadvisor.com (accessed on 21 January 2019).
7. Facebook. Available online: https://www.facebook.com (accessed on 21 January 2019).
8. Desrosiers, C.; Karypis, G. A Comprehensive Survey of Neighborhood-based Recommendation Methods. In *Recommender Systems Handbook*; Springer: New York, NY, USA, 2011; pp. 107–144.
9. Bobadilla, J.; Ortega, F.; Hernando, A.; Gutiérrez, A. Recommender Systems Survey. *Knowl.-Based Syst.* **2013**, *46*, 109–132. [CrossRef]
10. Adomavicius, G.; Tuzhilin, A. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 734–749. [CrossRef]

11. Li, Y.; Nie, J.; Zhang, Y. Contextual Recommendation Based on Text Mining. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters (COLING'10), Beijing, China, 23–27 August 2010; Association for Computational Linguistics: Stroudsburg, PA, USA, 2010; pp. 692–700.

12. Hariri, N.; Mobasher, B.; Burke, R.; Zheng, Y. Context-Aware Recommendation Based on Review Mining. In Proceedings of the 9th Workshop on Intelligent Techniques for Web Personalization and Recommender Systems (ITWP'11), Barcelona, Spain, 16 July 2011; CEUR-WS.org: Aachen, Germany, 2011; pp. 30–36.

13. Domingues, M.A.; Sundermann, C.V.; Manzato, M.G.; Marcacini, R.M.; Rezende, S.O. Exploiting Text Mining Techniques for Contextual Recommendations. In Proceedings of the IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies (WI '14), Warsaw, Poland, 11–14 August 2014; IEEE Computer Society: Washington, DC, USA, 2014; Volume 2, pp. 210–217.

14. Domingues, M.A.; Manzato, M.G.; Marcacini, R.M.; Sundermann, C.V.; Rezende, S.O. Using Contextual Information from Topic Hierarchies to Improve Context-Aware Recommender Systems. In Proceedings of the 22nd International Conference on Pattern Recognition (ICPR'14), Stockholm, Sweden, 24–28 August 2014; IEEE Computer Society: Washington, DC, USA, 2014; pp. 3606–3611.

15. Sundermann, C.; Domingues, M.; Marcacini, R.; Rezende, S. Using Topic Hierarchies with Privileged Information to Improve Context-Aware Recommender Systems. In Proceedings of the Brazilian Conference on Intelligent Systems (BRACIS'14), Sao Paulo, Brazil, 18–22 October 2014; IEEE Computer Society: Washington, DC, USA, 2014; pp. 61–66.

16. Sundermann, C.V.; Domingues, M.A.; Marcacini, R.M.; Rezende, S.O. Combining Privileged Information to Improve Context-Aware Recommender Systems. In Proceedings of the Encontro Nacional de Inteligência Artificial e Computacional (ENIAC'15), Natal, Brazil, 2015. Available online: https://arxiv.org/abs/1511.02290 (accessed on 21 January 2019).

17. Sundermann, C.V.; Domingues, M.A.; da Silva Conrado, M.; Rezende, S.O. Privileged contextual information for context-aware recommender systems. *Expert Syst. Appl.* **2016**, *57*, 139–158. [CrossRef]

18. Dey, A.K. Understanding and Using Context. *Pers. Ubiquitous Comput.* **2001**, *5*, 4–7. [CrossRef]

19. Chen, G.; Chen, L. Augmenting Service Recommender Systems by Incorporating Contextual Opinions from User Reviews. *User Model. User-Adapted Interact.* **2015**, *25*, 295–329. [CrossRef]

20. Chen, L.; Chen, G.; Wang, F. Recommender Systems Based on User Reviews: The State of the Art. *User Model. User-Adapted Interact.* **2015**, *25*, 99–154. [CrossRef]

21. Poriya, A.; Patel, N.; Bhagat, T.; Sharma, R. Non-Personalized Recommender Systems and User-based Collaborative Recommender Systems. *Int. J. Appl. Inf. Syst. (IJAIS)* **2014**, *6*, 22–27.

22. Hammar, M.; Karlsson, R.; Nilsson, B.J. Using Maximum Coverage to Optimize Recommendation Systems in e-Commerce. In Proceedings of the 7th ACM Conference on Recommender Systems (RecSys'13), Hong Kong, China, 12–16 October 2013; ACM: New York, NY, USA, 2013; pp. 265–272.

23. Schafer, J.B.; Frankowski, D.; Herlocker, J.; Sen, S. Collaborative Filtering Recommender Systems. In *The Adaptive Web*; Springer-Verlag: Berlin/Heidelberg, Germany, 2007; pp. 291–324.

24. Sparling, E.I.; Sen, S. Rating: How difficult is it? In Proceedings of the Fifth ACM Conference on Recommender Systems (RecSys '11), Chicago, IL, USA, 23–27 October 2011; ACM: New York, NY, USA, 2011; pp. 149–156.

25. Claypool, M.; Le, P.; Wased, M.; Browns, D. Implicit Interest Indicators. In Proceedings of the 6th International Conference on Intelligent User Interfaces (IUI'01), Santa Fe, NM, USA, 14–17 January 2001; ACM: New York, NY, USA, 2001; pp. 33–40.

26. Adomavicius, G.; Tuzhilin, A. Personalization Technologies: A Process-oriented Perspective. *Commun. ACM* **2005**, *48*, 83–90. [CrossRef]

27. Hill, W.; Stead, L.; Rosenstein, M.; Furnas, G. Recommending and Evaluating Choices in a Virtual Community of Use. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '95), Denver, CO, USA, 7–11 May 1995; ACM Press/Addison-Wesley Publishing Co.: New York, NY, USA, 1995; pp. 194–201. [CrossRef]

28. Konstan, J.A.; Miller, B.N.; Maltz, D.; Herlocker, J.L.; Gordon, L.R.; Riedl, J. GroupLens: Applying Collaborative Filtering to Usenet News. *Commun. ACM* **1997**, *40*, 77–87. [CrossRef]

29. Canny, J. Collaborative Filtering with Privacy via Factor Analysis. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02), Tampere, Finland, 11–15 August 2002; ACM: New York, NY, USA, 2002; pp. 238–245. [CrossRef]

30. Deshpande, M.; Karypis, G. Item-based top-N Recommendation Algorithms. *ACM Trans. Inf. Syst.* **2004**, *22*, 143–177. [CrossRef]

31. Bell, R.; Koren, Y.; Volinsky, C. Modeling Relationships at Multiple Scales to Improve Accuracy of Large Recommender Systems. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07), San Jose, CA, USA, 12–15 August 2007; ACM: New York, NY, USA, 2007; pp. 95–104. [CrossRef]

32. Koren, Y. Factorization Meets the Neighborhood: A Multifaceted Collaborative Filtering Model. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '08), Las Vegas, NV, USA, 24–27 August 2008; ACM: New York, NY, USA, 2008; pp. 426–434.

33. Abadi, M.J.; Luceri, L.; Hassan, M.; Chou, C.T.; Nicoli, M. A collaborative approach to heading estimation for smartphone-based PDR indoor localisation. In Proceedings of the 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Busan, South Korea, 27–30 October 2014; IEEE Computer Society: Washington, DC, USA, 2014; pp. 554–563. [CrossRef]

34. Abdelbar, A.M.; Elnabarawy, I.; Salama, K.M.; Wunsch, D.C. Matrix Factorization Based Collaborative Filtering With Resilient Stochastic Gradient Descent. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; IEEE Computer Society: Washington, DC, USA, 2018; pp. 1–7. [CrossRef]

35. Koren, Y. Collaborative Filtering with Temporal Dynamics. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09), Paris, France, 28 June–1 July 2009; ACM: New York, NY, USA, 2009; pp. 447–456.

36. Pazzani, M.; Billsus, D. Learning and Revising User Profiles: The Identification ofInteresting Web Sites. *Mach. Learn.* **1997**, *27*, 313–331. [CrossRef]

37. Billsus, D.; Pazzani, M.J. User Modeling for Adaptive News Access. *User Model. User-Adapted Interact.* **2000**, *10*, 147–180. [CrossRef]

38. Mooney, R.J.; Roy, L. Content-based Book Recommending Using Learning for Text Categorization. In Proceedings of the Fifth ACM Conference on Digital Libraries (DL '00), San Antonio, TX, USA, 2–7 June 2000; ACM: New York, NY, USA, 2000; pp. 195–204. [CrossRef]

39. Ahn, J.W.; Brusilovsky, P.; Grady, J.; He, D.; Syn, S.Y. Open User Profiles for Adaptive News Systems: Help or Harm? In Proceedings of the 16th International Conference on World Wide Web (WWW '07), Banff, AB, Canada, 8–12 May 2007; ACM: New York, NY, USA, 2007; pp. 11–20. [CrossRef]

40. Alanazi, A.; Bain, M. A People-to-people Content-based Reciprocal Recommender Using Hidden Markov Models. In Proceedings of the 7th ACM Conference on Recommender Systems (RecSys '13), Hong Kong, China, 12–16 October 2013; ACM: New York, NY, USA, 2013; pp. 303–306. [CrossRef]

41. Albatayneh, N.A.; Ghauth, K.I.; Chua, F.F. A Semantic Content-Based Forum Recommender System Architecture Based on Content-Based Filtering and Latent Semantic Analysis. In *Recent Advances on Soft Computing and Data Mining*; Herawan, T., Ghazali, R., Deris, M.M., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 369–378.

42. Adnan, M.N.M.; Chowdury, M.R.; Taz, I.; Ahmed, T.; Rahman, R.M. Content based news recommendation system based on fuzzy logic. In Proceedings of the 2014 International Conference on Informatics, Electronics Vision (ICIEV), Dhaka, Bangladesh, 23–24 May 2014; IEEE Computer Society: Washington, DC, USA, 2014; pp. 1–6. [CrossRef]

43. Balabanovic, M.; Shoham, Y. Fab: Content-based, Collaborative Recommendation. *Commun. ACM* **1997**, *40*, 66–72. [CrossRef]

44. Ahmad Wasfi, A.M. Collecting User Access Patterns for Building User Profiles and Collaborative Filtering. In Proceedings of the 4th International Conference on Intelligent User Interfaces (IUI '99), Los Angeles, CA, USA, 5–8 January 1999; ACM: New York, NY, USA, 1999; pp. 57–64. [CrossRef]

45. Claypool, M.; Gokhale, A.; Miranda, T.; Murnikov, P.; Netes, D.; Sartin, M. Combining Content-Based and Collaborative Filters in an Online Newspaper. In Proceedings of the ACM SIGIR Workshop on Recommender Systems, Berkeley, CA, USA, 1 September 1999; ACM: New York, NY, USA, 1999.

46. Melville, P.; Mooney, R.J.; Nagarajan, R. Content-boosted Collaborative Filtering for Improved Recommendations. In Proceedings of the Eighteenth National Conference on Artificial Intelligence, Edmonton, AB, Canada, 28 July–1 August 2002; American Association for Artificial Intelligence: Menlo Park, CA, USA, 2002; pp. 187–192.

47. De Campos, L.M.; Fernández-Luna, J.M.; Huete, J.F.; Rueda-Morales, M.A. Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks. *Int. J. Approx. Reason.* **2010**, *51*, 785–799. [CrossRef]

48. Akehurst, J.; Koprinska, I.; Yacef, K.; Pizzato, L.A.S.; Kay, J.; Rej, T. CCR—A Content-Collaborative Reciprocal Recommender for Online Dating. In Proceedings of the Twenty-Second international joint conference on Artificial Intelligence (IJCAI '11), Barcelona, Catalonia, Spain, 16–22 July 2011; AAAI Press: Palo Alto, CA, USA, 2011.

49. Cremonesi, P.; Turrin, R.; Airoldi, F. Hybrid Algorithms for Recommending New Items. In Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec '11), Chicago, IL, 27 October 2011; ACM: New York, NY, USA, 2011; pp. 33–40. [CrossRef]

50. Aguilar, J.; Portilla, O.; Puerto, E. Adaptive hybrid recommender system of learning objects. In Proceedings of the 2016 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Cartagena, Colombia, 2–4 November 2016; IEEE Computer Society: Washington, DC, USA, 2016; pp. 1–6. [CrossRef]

51. Agrawal, M.; Gonçalves, T.; Quaresma, P. A hybrid approach for cold start recommendations. In Proceedings of the 2017 11th International Conference on Software, Knowledge, Information Management and Applications (SKIMA), Malabe, Sri Lanka, 6–8 December 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 1–6. [CrossRef]

52. Adomavicius, G.; Sankaranarayanan, R.; Sen, S.; Tuzhilin, A. Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach. *ACM Trans. Inf. Syst.* **2005**, *23*, 103–145. [CrossRef]

53. Domingues, M.A.; Jorge, A.M.; Soares, C. Dimensions As Virtual Items: Improving the Predictive Ability of top-N Recommender Systems. *Inf. Process. Manag. Int. J.* **2013**, *49*, 698–720. [CrossRef]

54. Adomavicius, G.; Tuzhilin, A. Context-Aware Recommender Systems. In *Recommender Systems Handbook*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 217–253.

55. Gauch, S.; Speretta, M.; Chandramouli, A.; Micarelli, A. User Profiles for Personalized Information Access. In *The Adaptive Web: Methods and Strategies of Web Personalization*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 54–89. [CrossRef]

56. Pouli, V.; Kafetzoglou, S.; Tsiropoulou, E.E.; Dimitriou, A.; Papavassiliou, S. Personalized multimedia content retrieval through relevance feedback techniques for enhanced user experience. In Proceedings of the 2015 13th International Conference on Telecommunications (ConTEL), Graz, Austria, 13–15 July 2015; IEEE Computer Society: Washington, DC, USA; pp. 1–8. [CrossRef]

57. Amato, F.; Moscato, V.; Picariello, A.; Sperlí, G. KIRA: A System for Knowledge-Based Access to Multimedia Art Collections. In Proceedings of the 2017 IEEE 11th International Conference on Semantic Computing (ICSC), San Diego, CA, USA, 30 January–1 February2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 338–343. [CrossRef]

58. Stai, E.; Kafetzoglou, S.; Tsiropoulou, E.E.; Papavassiliou, S. A Holistic Approach for Personalization, Relevance Feedback & Recommendation in Enriched Multimedia Content. *Multimedia Tools Appl.* **2018**, *77*, 283–326. [CrossRef]

59. Persico, V.; Pescapé, A.; Picariello, A.; Sperlí, G. Benchmarking big data architectures for social networks data processing using public cloud platforms. *Future Gener. Comput. Syst.* **2018**, *89*, 98–109. [CrossRef]

60. Panniello, U.; Gorgoglione, M. Incorporating Context into Recommender Systems: An Empirical Comparison of Context-based Approaches. *Eletron. Commer. Res.* **2012**, *12*, 1–30. [CrossRef]

61. IMDb. Available online: https://www.imdb.com (accessed on 21 January 2019).

62. Tang, H.; Tan, S.; Cheng, X. A survey on sentiment detection of reviews. *Expert Syst. Appl.* **2009**, *36*, 10760–10773. [CrossRef]

63. Liu, B. *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*; Cambridge University Press: Cambridge, UK, 2015.

64. Ganu, G.; Elhadad, N.; Marian, A. Beyond the Stars: Improving Rating Predictions using Review Text Content. In Proceedings of the 12th International Workshop on the Web and Databases (WebDB'09), Providence, RI, USA, 28 June 2009; pp. 1–6.

65. Dave, K.; Lawrence, S.; Pennock, D.M. Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. In Proceedings of the 12th International Conference on World Wide Web (WWW'03), Budapest, Hungary, 20–24 May 2003; ACM: New York, NY, USA, 2003; pp. 519–528.

66. Pang, B.; Lee, L.; Vaithyanathan, S. Thumbs Up?: Sentiment Classification Using Machine Learning Techniques. In Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing (EMNLP'02), Philadelphia, PA, USA, 6–7 July 2002; Association for Computational Linguistics: Stroudsburg, PA, USA, 2002; Volume 10, pp. 79–86.

67. Wiebe, J. Learning Subjective Adjectives from Corpora. In Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence (AAAI '00), Austin, TX, USA, 30 July–3 August 2000; AAAI Press: Palo Alto, CA, USA, 2000; pp. 735–740.

68. Liu, B. Sentiment Analysis and Opinion Mining. *Synth. Lect. Hum. Lang. Technol.* **2012**, *5*, 1–167. [CrossRef]

69. Liu, B. Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data. In *Data-Centric Systems and Applications*; Springer-Verlag New York, Inc.: New York, NY, USA, 2006.

70. Hu, M.; Liu, B. Mining and Summarizing Customer Reviews. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'04), Seattle, WA, USA, 22–25 August 2004; ACM: New York, NY, USA, 2004; pp. 168–177.

71. Popescu, A.M.; Etzioni, O. Extracting Product Features and Opinions from Reviews. In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT'05), Vancouver, BC, Canada, 6–8 October 2005; Association for Computational Linguistics: Stroudsburg, PA, USA, 2005; pp. 339–346.

72. Long, C.; Zhang, J.; Zhut, X. A Review Selection Approach for Accurate Feature Rating Estimation. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters (COLING'10), Beijing, China, 23–27 August 2010; Association for Computational Linguistics: Stroudsburg, PA, USA, 2010; pp. 766–774.

73. Blair-Goldensohn, S.; Hannan, K.; McDonald, R.; Neylon, T.; Reis, G.A.; Reynar, J. Building a sentiment summarizer for local service reviews. In Proceedings of the workshop on challenges in the Information Explosion Era (WWW'08), Beijing, China, 22 April 2008; ACM: New York, NY, USA, 2008; Volume 14.

74. Lafferty, J.D.; McCallum, A.; Pereira, F.C.N. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In Proceedings of the Eighteenth International Conference on Machine Learning (ICML'01), Williamstown, MA, USA, 28 June–2 July 2001; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2001; pp. 282–289.

75. Rabiner, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **1989**, *77*, 257–286. [CrossRef]

76. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.

77. Hofmann, T. Probabilistic Latent Semantic Indexing. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99), Berkeley, CA, USA, 15–19 August 1999; ACM: New York, NY, USA, 1999; pp. 50–57.

78. Ding, X.; Liu, B.; Yu, P.S. A holistic lexicon-based approach to opinion mining. In Proceedings of the Conference on Web Search and Web Data Mining (WSDM'08), Palo Alto, CA, USA, 11–12 February 2008; ACM: New York, NY, USA, 2008.

79. Liu, B. Sentiment Analysis and Subjectivity. In *Handbook of Natural Language Processing*, 2nd ed.; Indurkhya, N., Damerau, F.J., Eds.; Taylor & Francis Group: Abingdon, UK, 2010.

80. Kitchenham, B.; Charters, S. *Guidelines for Performing Systematic Literature Reviews in Software Engineering*; Technical Report, EBSE Technical Report EBSE-2007-01, Keele University and Durham University Joint Report; Keele University: Keele, UK, 2007.

81. Portugal, I.; Alencar, P.S.C.; Cowan, D.D. The Use of Machine Learning Algorithms in Recommender Systems: A Systematic Review. *CoRR* **2015**, *97*, 205–227. [CrossRef]

82. Rahayu, P.; Sensuse, D.I.; Purwandari, B.; Budi, I.; Khalid, F.; Zulkarnaim, N. A Systematic Review of Recommender System for e-Portfolio Domain. In Proceedings of the 5th International Conference on Information and Education Technology (ICIET '17), Tokyo, Japan, 10–12 January 2017; ACM: New York, NY, USA, 2017; pp. 21–26.

83. Kumar, B.; Sharma, N. Approaches, Issues and Challenges in Recommender Systems: A Systematic Review. *Indian J. Sci. Technol.* **2016**. [CrossRef]

84. Sezgin, E.; Özkan, S. A systematic literature review on Health Recommender Systems. In Proceedings of the 2013 E-Health and Bioengineering Conference (EHB), Iasi, Romania, 21–23 November 2013; IEEE Computer Society: Washington, DC, USA; pp. 1–4.

85. Scopus. Available online: https://www.scopus.com (accessed on 21 January 2019).

86. ACM Digital Library. Available online: http://dl.acm.org (accessed on 21 January 2019).

87. IEEE Xplore Digital Library. Available online: http://ieeexplore.ieee.org/Xplore/home.jsp (accessed on 21 January 2019).

88. ScienceDirect. Available online: http://www.sciencedirect.com (accessed on 21 January 2019).

89. Moher, D.; Liberati, A.; Tetzlaff, J.; Altman, D.G.; Group, T.P. Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement. *PLoS Med.* **2009**, *6*, e1000097. [CrossRef]

90. Ho, S.S.; Lieberman, M.; Wang, P.; Samet, H. Mining Future Spatiotemporal Events and Their Sentiment from Online News Articles for Location-aware Recommendation System. In Proceedings of the First ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems (MobiGIS'12), Redondo Beach, CA, USA, 6–9 November 2012; ACM: New York, NY, USA, 2012; pp. 25–32.

91. Levi, A.; Mokryn, O.; Diot, C.; Taft, N. Finding a Needle in a Haystack of Reviews: Cold Start Context-based Hotel Recommender System. In Proceedings of the Sixth ACM Conference on Recommender Systems (RecSys'12), Dublin, Ireland, 9–13 September 2012; ACM: New York, NY, USA, 2012; pp. 115–122.

92. Meehan, K.; Lunney, T.; Curran, K.; McCaughey, A. Context-aware intelligent recommendation system for tourism. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications (Workshops) (PERCOM'13), San Diego, CA, USA, 18–22 March 2013; IEEE Computer Society: Washington, DC, USA, 2013; pp. 328–331.

93. Chen, G.; Chen, L. Recommendation Based on Contextual Opinions. In *User Modeling, Adaptation, and Personalization*; Springer International Publishing: Berlin/Heidelberg, Germany, 2014; Volume 8538, pp. 61–73.

94. Colace, F.; Santo, M.D.; Greco, L.; Moscato, V.; Picariello, A. A collaborative user-centered framework for recommending items in Online Social Networks. *Comput. Hum. Behavior* **2015**, *51 Part B*, 694–704. [CrossRef]

95. Colace, F.; De Santo, M.; Greco, L.; Amato, F.; Moscato, V.; Persia, F.; Picariello, A. A user-centered approach for social recommendations. In Proceedings of the Eighth International Conference on Advances in Computer-Human Interactions (ACHI'15), Venice, Italy, 22–27 February 2015; IEEE Computer Society: Washington, DC, USA, 2015; pp. 190–193.

96. Kothari, A.A.; Patel, W.D. A Novel Approach Towards Context Sensitive Recommendations Based on Machine Learning Methodology. In Proceedings of the Fifth International Conference on Communication Systems and Network Technologies (CSNT'15), Gwalior, India, 4–6 April 2015; IEEE Computer Society: Washington, DC, USA, 2015; pp. 1114–1118.

97. Orellana-Rodriguez, C.; Diaz-Aviles, E.; Nejdl, W. Mining Affective Context in Short Films for Emotion-Aware Recommendation. In Proceedings of the 26th ACM Conference on Hypertext and Social Media (HT'15), Guzelyurt, Northern Cyprus, 2–4 September 2015; ACM: New York, NY, USA, 2015; pp. 185–194.

98. Yang, P.; Wang, H.; Fang, H.; Cai, D. Opinions Matter: A General Approach to User Profile Modeling for Contextual Suggestion. *Inf. Retr.* **2015**, *18*, 586–610. [CrossRef]

99. Zhao, G.; Qian, X.; Lei, X.; Mei, T. Service Quality Evaluation by Exploring Social Users' Contextual Information. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 3382–3394. [CrossRef]

100. Kharrat, F.B.; Elkhleifi, A.; Faiz, R. Recommendation system based contextual analysis of Facebook comment. In Proceedings of the 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), Agadir, Morocco, 29 November–2 December 2016; IEEE Computer Society: Washington, DC, USA, 2016; pp. 1–6.

101. Missaoui, S.; Viviani, M.; Faiz, R.; Pasi, G. A Language Modeling Approach for the Recommendation of Tourism-related Services. In Proceedings of the Symposium on Applied Computing (SAC '17), Marrakech, Morocco, 3–7 April 2017; ACM: New York, NY, USA, 2017; pp. 1697–1700.

102. Jalan, K.; Gawande, K. Context-aware hotel recommendation system based on hybrid approach to mitigate cold-start-problem. In Proceedings of the International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS '17), Chennai, India, 1–2 August 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 2364–2370. [CrossRef]

103. Baral, R.; Zhu, X.; Iyengar, S.S.; Li, T. ReEL: R Eview Aware Explanation of Location Recommendation. In Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization (UMAP '18), Singapore, 8–11 July 2018; ACM: New York, NY, USA, 2018; pp. 23–32. [CrossRef]

104. Sulthana, A.R.; Ramasamy, S. Ontology and context based recommendation system using Neuro-Fuzzy Classification. *Comput. Electr. Eng.* **2018**. [CrossRef]

105. Zangerle, E.; Chen, C.; Tsai, M.; Yang, Y. Leveraging Affective Hashtags for Ranking Music Recommendations. *IEEE Trans. Affect. Comput.* **2018**. [CrossRef]

106. Reichardt, J.; Bornholdt, S. Statistical mechanics of community detection. *Phys. Rev. E* **2006**, *74*, 016110. [CrossRef] [PubMed]

107. Wang, H.; Lu, Y.; Zhai, C. Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'10), Washington, DC, USA, 24–28 July 2010; ACM: New York, NY, USA, 2010; pp. 783–792.

108. Colace, F.; De Santo, M.; Greco, L. A Probabilistic Approach to Tweets' Sentiment Classification. In Proceedings of the Humaine Association Conference on Affective Computing and Intelligent Interaction (ACII'13), Geneva, Switzerland, 2–5 September 2013; IEEE Computer Society: Washington, DC, USA, 2013; pp. 37–42.

109. Zhang, W.; Ding, G.; Chen, L.; Li, C.; Zhang, C. Generating Virtual Ratings from Chinese Reviews to Augment Online Recommendations. *ACM Trans. Intell. Syst. Technol.* **2013**, *4*, 9. [CrossRef]

110. GeoNames gazetteer. Available online: http://www.geonames.org (accessed on 21 January 2019).

111. Miller, G.A.; Beckwith, R.; Fellbaum, C.; Gross, D.; Miller, K.J. Introduction to WordNet: An on-line lexical database. *Int. J. Lexicogr.* **1990**, *3*, 235–244. [CrossRef]

112. Amazon Mechanical Turk Workers. Available online: https://www.mturk.com (accessed on 21 January 2019).

113. Hayes, C.; Massa, P.; Avesani, P.; Cunningham, P. An On-Line Evaluation Framework for Recommender Systems. In Proccedings of the Workshop on Personalization and Recommendation in E-Commerce (AH'02), Malaga, Spain, 28 May 2002; Springer Verlag: Berlin/Heidelberg, Germany, 2002.

114. WorldWeatherOnline API. Available online: https://developer.worldweatheronline.com/api (accessed on 21 January 2019).

115. Yang, Y.; Pedersen, J.O. A Comparative Study on Feature Selection in Text Categorization. In Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97), Nashville, TN, USA, 8–12 July 1997; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1997; pp. 412–420.

116. Wilson, T. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT'05), Vancouver, BC, Canada, 6–8 October 2005; Association for Computational Linguistics: Stroudsburg, PA, USA, 2005; pp. 347–354.

117. Yelp. Available online: https://www.yelp.com/ (accessed on 21 January 2019).

118. Shani, G.; Gunawardana, A. Evaluating Recommender Systems. In *DS'12: Recommender Systems Handbook*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 257–298.

119. Codina, V.; Ricci, F.; Ceccaroni, L. Exploiting the Semantic Similarity of Contextual Situations for Pre-filtering Recommendation. In *User Modeling, Adaptation, and Personalization*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2013; Volume 7899, pp. 165–177.

120. Adomavicius, G.; Kwon, Y. New recommendation techniques for multicriteria rating systems. *IEEE Intell. Syst.* **2007**, *22*, 48–55. [CrossRef]

121. Esuli, A.; Sebastiani, F. SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining. In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06), Genoa, Italy, 22–28 May 2006; pp. 417–422.

122. Neviarouskaya, A.; Prendinger, H.; Ishizuka, M. SentiFul: A Lexicon for Sentiment Analysis. *IEEE Trans. Affect. Comput.* **2011**, *2*, 22–36. [CrossRef]

123. Su, X.; Khoshgoftaar, T.M. A Survey of Collaborative Filtering Techniques. *Adv. Artif. Intell.* **2009**, *2009*, 421425. [CrossRef]

124. Youtube. Available online: https://www.youtube.com (accessed on 21 January 2019).

125. Youtube's Data API. Available online: https://developers.google.com/youtube/v3/ (accessed on 21 January 2019).

126. LingPipe. Available online: http://alias-i.com/lingpipe (accessed on 21 January 2019).

127. MorphAdorner. Available online: http://morphadorner.northwestern.edu (accessed on 21 January 2019).

128. Ganesan, K.; Zhai, C.; Han, J. Opinosis: A Graph-based Approach to Abstractive Summarization of Highly Redundant Opinions. In Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10), Beijing, China, 23–27 August 2010; Association for Computational Linguistics: Stroudsburg, PA, USA, 2010; pp. 340–348.

129. Yang, P.; Fang, H. An exploration of ranking-based strategy for contextual suggestion. In Proceedings of the Text Retrieval Conference (TREC'12), Gaithersburg, MD, USA, 6–9 November 2012; National Institute of Standards and Technology (NIST): Gaithersburg, MD, USA, 2012.

130. Koenigstein, N.; Dror, G.; Koren, Y. Yahoo! Music Recommendations: Modeling Music Ratings with Temporal Dynamics and Item Taxonomy. In Proceedings of the Fifth ACM Conference on Recommender Systems (RecSys '11), Chicago, IL, USA, 23–27 October 2011; ACM: New York, NY, USA, 2011; pp. 165–172.

131. Salakhutdinov, R.; Mnih, A. Probabilistic matrix factorization. In Proceedings of the Advances in Neural Information Processing Systems (NIPS'07), Vancouver, BC, Canada, 3–6 December 2007; Curran Associates Inc.: Red Hook, NY, USA, 2007; pp. 1257–1264.

132. Yang, X.; Steck, H.; Liu, Y. Circle-based Recommendation in Online Social Networks. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '12), Beijing, China, 12–16 August 2012; ACM: New York, NY, USA, 2012; pp. 1267–1275.

133. Jiang, M.; Cui, P.; Liu, R.; Yang, Q.; Wang, F.; Zhu, W.; Yang, S. Social Contextual Recommendation. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12), Maui, HI, USA, 29 October–2 November 2012; ACM: New York, NY, USA, 2012; pp. 45–54.

134. Feng, H.; Qian, X. Recommendation via User's Personality and Social Contextual. In Proceedings of the 22nd ACM International Conference on Information & Knowledge Management (CIKM '13), San Francisco, CA, USA, 27 October–1 November 2013; ACM: New York, NY, USA, 2013; pp. 1521–1524.

135. Qian, X.; Feng, H.; Zhao, G.; Mei, T. Personalized Recommendation Combining User Interest and Social Circle. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 1763–1777. [CrossRef]

136. Sarwar, B.; Karypis, G.; Konstan, J.; Riedl, J. Item-based Collaborative Filtering Recommendation Algorithms. In Proceedings of the 10th International Conference on World Wide Web (WWW '01), Hong Kong, China, 1–5 May 2001; ACM: New York, NY, USA, 2001; pp. 285–295.

137. Friedman, J.H.; Meulman, J.J. Multiple additive regression trees with application in epidemiology. *Stat. Med.* **2003**, *22*, 1365–1381. [CrossRef]

138. Bellogin, A.; Castells, P.; Cantador, I. Precision-oriented Evaluation of Recommender Systems: An Algorithmic Comparison. In Proceedings of the Fifth ACM Conference on Recommender Systems (RecSys '11), Chicago, IL, USA, 23–27 October 2011; ACM: New York, NY, USA, 2011; pp. 333–336.

139. OpenNLP. Available online: https://opennlp.apache.org/ (accessed on 21 January 2019).

140. AirBnB. Available online: https://www.airbnb.com (accessed on 21 January 2019).

141. Stanford Parser. Available online: https://nlp.stanford.edu/software/lex-parser.shtml (accessed on 21 January 2019).

142. SentiWordNet. Available online: https://sentiwordnet.isti.cnr.it/ (accessed on 21 January 2019).

143. Aciar, S.; Zhang, D.; Simoff, S.; Debenham, J. Informed Recommender: Basing Recommendations on Consumer Product Reviews. *IEEE Intell. Syst.* **2007**, *22*, 39–47. [CrossRef]

144. Nielsen, F.R. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. *arXiv* **2011**, arXiv:1103.2903.

145. Thelwall, M.; Buckley, K.; Paltoglou, G.; Cai, D.; Kappas, A. Sentiment Strength Detection in Short Informal Text. *J. Am. Soc. Inf. Sci. Technol.* **2010**, *61*, 2544–2558. [CrossRef]

146. Hutto, C.J.; Gilbert, E. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. In Proceedings of the International AAAI Conference on Web and Social Media (ICWSM '14), Ann Arbor, MI, USA, 1–4 June 2014; AAAI Press: Palo Alto, CA, USA, 2014.